15th European Conference on Object-Oriented Programming

Budapest, Hungary, June 18-22, 2001

Organized by

AitO

Main sponsors:

IQSOFT the e-developers' company

Ministry of Education

15. ECOOP 2001 BUDAPEST

www.2001.ecoop.org

Welcome!

Welcome by the Organizing Chairs of the 15th European Conference on Object-Oriented Programming. We cordially invite all researchers, practitioners, educators and students, everyone interested in object technology to attend, whether they come from academia, industry or any other sector of society. ECOOP demonstrates the very best and latest research, practice and experience using object technologies.

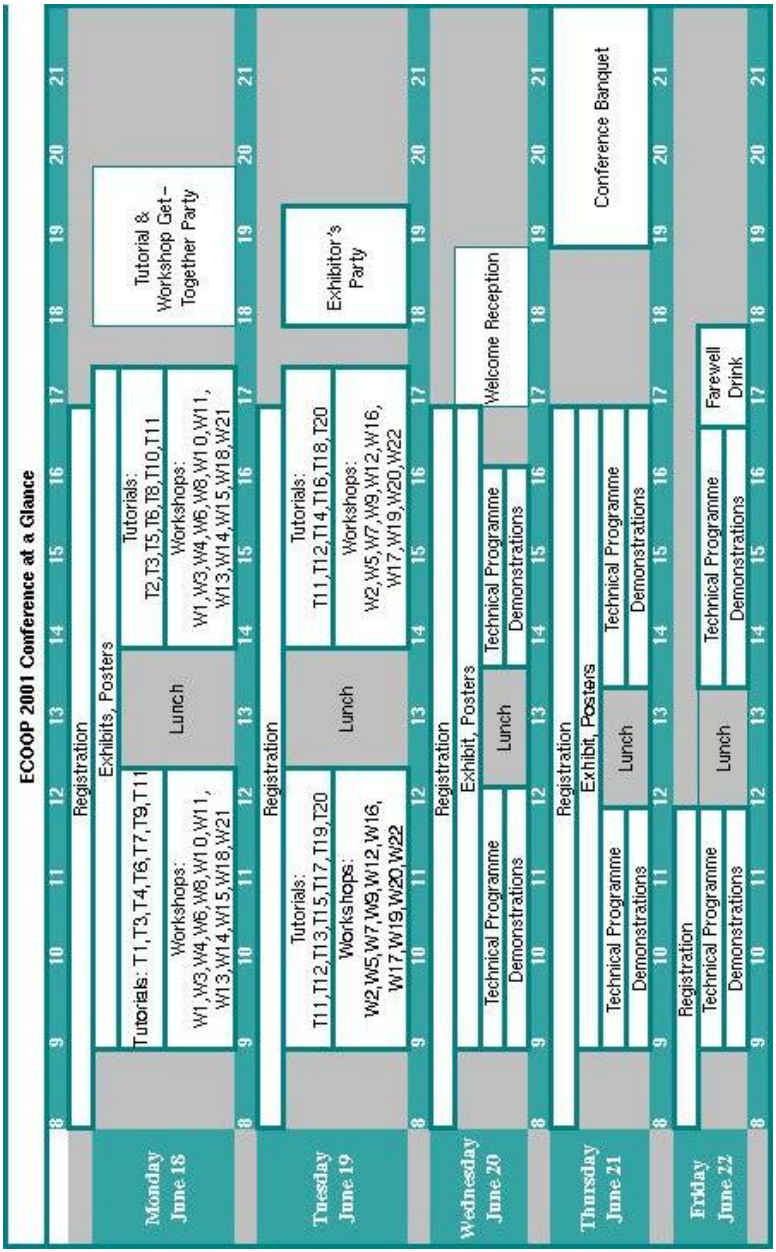
The conference is hosted by Eötvös Loránd University, Budapest, Hungary (supported by the Association Internationale pour les Technologies Objets) and will run from June 18 to June 22, 2001. We follow the traditions of ECOOP, so a number of tutorials and workshops are planned for the first two days and technical presentations, panel discussions, poster presentations and invited talks will take place on the last three days of the conference. Simultaneously there will also be an exhibitor's programme allowing object-oriented products and techniques to be displayed and demonstrated.

The conference site is Budapest, the capital of Hungary. It is located on the river Danube, in the heart of Central Europe. The participants will be lodged in the centre of Budapest. The conference hosts different social events and thus offers participants lots of opportunities for a casual get-together. The social programme will give the participants an opportunity to get acquainted with Hungarian culture and history since this year is the one thousand years anniversary of the founding of the Hungarian state.

Organizing Chairs

Up-to-date information is available at **www.2001.ecoop.org**

Table of Contents


ECOOP 2001 Conference at a Glance
8 9 10 11 12 13 14 15 16 17 18 19 20 21
Monday June 18
Registration
Exhibits, Posters
Tutorials: T1,T3,T4,T6,T7,T9,T11
Workshops: W1,W3,W4,W6,W8,W10,W11, W13,W14,W15,W18,W21
Lunch
Tutorials: T2,T3,T5,T6,T8,T10,T11
Workshops: W1,W3,W4,W6,W8,W10,W11, W13,W14,W15,W18,W21
Tutorial & Workshop Get-Together Party
Tuesday June 19
Registration
Tutorials: T11,T12,T13,T15,T17,T19,T20
Workshops: W2,W5,W7,W9,W12,W16, W17,W19,W20,W22
Lunch
Tutorials: T11,T12,T14,T16,T18,T20
Workshops: W2,W5,W7,W9,W12,W16, W17,W19,W20,W22
Exhibitor's Party
Wednesday June 20
Registration
Exhibit, Posters
Technical Programme
Demonstrations
Lunch
Technical Programme
Demonstrations
Welcome Reception
Thursday June 21
Registration
Exhibit, Posters
Technical Programme
Demonstrations
Lunch
Technical Programme
Demonstrations
Conference Banquet
Friday June 22
Registration
Technical Programme
Demonstrations
Lunch
Technical Programme
Demonstrations
Farewell Drink

Executive Committee

Conference Chairs:	*Gerti Kappel*, Johannes Kepler University of Linz, Austria *László Varga*, Eötvös Loránd University, Budapest, Hungary
Programme Chair:	*Jørgen Lindskov Knudsen*, University of Aarhus, Denmark
Organizing Chairs:	*László Kozma*, Eötvös Loránd University, Budapest, Hungary *Zoltán Horváth*, Eötvös Loránd University, Budapest, Hungary

Programme Committee

Mehmet Aksit	*University of Twente, The Netherlands*
Suad Alagic	*Wichita State University, USA*
Elisa Bertino	*University of Milan, Italy*
Gilad Bracha	*Sun Microsystems, USA*
Jean-Pierre Briot	*University Paris 6 - CNRS, France*
Frank Buschmann	*Siemens AG, Corporate Technology, Germany*
Giuseppe Castagna	*École Normale Supérieure de Paris, France*
Pierre Cointe	*École des Mines de Nantes, France*
Serge Demeyer	*University of Antwerp, Belgium*
Peter Dickman	*University of Glasgow, UK*
Bjørn N. Freeman-Benson	*Quicksilver technology, USA*
Carlo Ghezzi	*Politecnico di Milano, Italy*
Joseph Gil	*Technion, Israel Inst. of Technology, Israel*
Rachid Guerraoui	*Swiss Federal Institute of Technology, Switzerland*
Görel Hedin	*Lund University, Sweden*
Urs Hölzle	*University of California at Santa Barbara, USA*
Eric Jul	*University of Copenhagen, Denmark*
Gerti Kappel	*Johannes Kepler University of Linz, Austria*
Jørgen Lindskov Knudsen	*University of Aarhus, Denmark*

Karl Lieberherr	*Northeastern University, USA*
Cristina Videira Lopes	*Xerox Palo Alto Research Center, USA*
Satoshi Matsuoka	*Tokyo Institute of Technology, Japan*
Mira Mezini	*Technical University of Darmstadt, Germany*
Ana Moreira	*Universidade Nova de Lisboa, Portugal*
James Noble	*Victoria University of Wellington, New Zealand*
Rui Oliveira	*Universidade de Minho, Portugal*
Jens Palsberg	*Purdue University, USA*
Markku Sakkinen	*Tampere University of Technology, Finland*
Dag Sjøberg	*University of Oslo, Norway*
Jan Vitek	*Purdue University, USA*
John Vlissides	*IBM T. J. Watson Research Center, USA*

Organizing Committee

Tutorial Chairs:	*Judit Nyéky-Gaizler, Eötvös Loránd University, Budapest, Hungary*
	Giuseppe Castagna, École Normale Supérieure, Paris, France
	Katalin Pásztor-Varga, Eötvös Loránd University, Budapest, Hungary
Workshop Chairs:	*Ákos Frohner, Eötvös Loránd University, Budapest, Hungary*
	Jacques Malenfant, Université de Bretagne sud, Vannes, France
	Erzsébet Angster, Dennis Gabor College, Budapest, Hungary
Panel Chair:	*Tamás Kozsik, Eötvös Loránd University, Budapest, Hungary*
Poster Chairs:	*Viktória Zsók, Eötvös Loránd University, Budapest, Hungary*
	István Juhász, University of Debrecen, Hungary
Demonstration Chairs:	*János Vida, Eötvös Loránd University, Budapest, Hungary*
	Károly Kondorosi, Technical University of Budapest, Hungary
Exhibit Chairs:	*Zoltán Csörnyei, Eötvös Loránd University, Budapest, Hungary*
	László Blum, University of Veszprém, Hungary

Social Events:	*Rozália Szabó-Nacsa, Eötvös Loránd University, Budapest, Hungary*
Advertizing, Web site:	*Zoltán Istenes, Eötvös Loránd University, Budapest, Hungary*
	Attila Kovács, Eötvös Loránd University, Budapest, Hungary
Industrial Relations Chairs:	*Bálint Dömölki, IQSOFT Rt., Budapest, Hungary*
	Albert Fazekas, Bull Hungary, Budapest, Hungary
	Miklós Havass, SZÁMALK Rt., Budapest, Hungary
Operational Support	*Zoltán Porkoláb, Eötvös Loránd University, Budapest, Hungary*
	Ferenc Steingart, Eötvös Loránd University, Budapest, Hungary

Sponsors

ECOOP 2001 received financial support from public institutions as well as from industrial companies. The organizers would like to express here their gratitude to these sponsors.

Public Institutions and Non–profit Organizations

Eötvös Loránd University, Ministry of Education

Industrial Companies

Platinum level:	IQSoft*, ComputerWorld
Gold level:	Graphisoft, Microsoft
Silver level:	Ericsson, ERCIM
Bronze level:	Nokia, VT-Soft, Számalk

*IQSoft is the main professional sponsor of ECOOP 2001.

Tutorials

Each tutorial lasts either one half day or one full day (except for T11, which lasts two full days). Tutorials take place on Monday and Tuesday in the Northern Block of the ELTE University. They will be held from 9:00 to 12:30 and from 14:00 to 17:30. More up-to-date information about the tutorials is available on our website.

Tutorial timetable

	Title	*Mon*		*Tue*	
		am	*pm*	*am*	*pm*
T1	Integrating Business Objects Into J2EE *Martin Fowler*	☒			
T2	Making the Software Process Transparent by Using Intelligent Agents *Ivar Jacobson, Gunnar Övergaard*		☒		
T3	Concurrent Object-Oriented Design in Java *Doug Lea, David Holmes*	☒	☒		
T4	Design of Software Architectures *Jan Bosch*	☒			
T5	Enterprise JavaBean Architecture and Design Issues; Avoiding JavaBean Soup *James White*		☒		
T6	Generative Programming: Methods, Techniques, and Applications *Krzysztof Czarnecki, Ulrich W. Eisenecker*	☒	☒		
T7	Adapting XP to Complex Application Domains *Martin Lippert, Stefan Roock*	☒			
T8	Object-Oriented Frameworks *Greg Butler*		☒		
T9	Programming with XML *Peter Bunemann*	☒			

	Title	Mon		Tue	
		am	pm	am	pm
T10	Squeak: An Open Source Smalltalk for the 21st Century! *Andrew P. Black*		☒		
T11	Pattern Mastery through Pattern Writing *James O. Coplien, Christa Schwanninger*	☒	☒	☒	☒
T12	Aspect-Oriented Programming with AspectJ™ *Erik Hilsdale, Gregor Kiczales*			☒	☒
T13	Patterns at Work *Frank Buschmann*			☒	
T14	Advanced Visual Modelling: Beyond UML *Joseph (Yossi) Gil, John Howse, Stuart Kent*				☒
T15	Interconnecting Objects via Contracts *José Luiz Fiadeiro, Luís Filipe Andrade*			☒	
T16	Advanced Software Composition: Obstacles & Approaches *Lodewijk Bergmans, Mehmet Aksit*				☒
T17	Introduction to Java-based Mobile Agent Programming Systems *Anand Tripathi*			☒	
T18	Developing applications for the J2ME™ Mobile Information Device Profile *Aldo Eisma, Michael Brung*				☒
T19	Component Technologies for the Middle-Tier *Michael Stal*			☒	
T20	Real Time System Modelling with UML: Current Status and Some Prospects *François Terrier, Sébastien Gerard*			☒	☒
T21	Security and Object-Oriented Programming *Dieter Gollmann*				☒

T01: Integrating Business Objects Into J2EE

Presenter: Martin Fowler

Duration: half day

Level: Intermediate

Day: Monday (June 18) a.m.

The Java 2 Enterprise Edition platform (J2EE) - with its technologies JSP, Servlets, EJB, and JDBC - is rapidly becoming a popular platform for developing business software. Developers are interested in its portability, features, and a good range of vendors that support its standards. However it is still new technology and everyone is still learning about how best to use it. ThoughtWorks have a taste for the bleeding edge and over the last year or so they've been building a few serious J2EE systems for their clients. As a result we've learned first hand some important lessons about using this technology. In this talk we'll go over the decisions, patterns, and pitfalls you'll run into if you're looking to build a J2EE system. We'll discuss various server architectures for J2EE, how (and whether) to use Entity Beans, communication between client and server, remote interfaces, alternative client UIs, and database integration.

Audience: Architects and software developers.

Required experience: Attendees should have a working knowledge of Java.

Presenter's profile: Martin Fowler is the Chief Scientist for ThoughtWorks, Inc., an Internet professional services provider specializing in the deliver of highly strategic B2B e-Commerce solutions. As an independent consultant he's spent over a decade applying object technology to business information systems mentoring clients in business object development, analysis and design techniques, UML, Refactoring, lightweight development processes, and the use of patterns. He is the author of Analysis Patterns, Refactoring, Planning Extreme Programming, and the award winning UML Distilled.

T02: Making the Software Process Transparent by Using Intelligent Agents

Presenters: Ivar Jacobson, Gunnar Övergaard

Duration: half day

Level: Intermediate

Day: Monday (June 18) p.m.

It has never been so hard to develop good software as today. Developers need more knowledge and skill than ever before. They need to be skilled in programming languages (e.g. Java, C++), system software platforms (.NET, J2EE), XML, middleware (WebSphere, Logicworks, etc.), the Unified Modelling Language, the Rational Unified Process, web architectures, etc. And they need to learn about these technologies faster than ever with almost no time for training and education. If they don't their only rescue is to find shortcuts, use lightweight methodologies and ignoring well-proven best practices. And as usual quality will suffer.

There is another way. In this tutorial we will discuss how software agents can be used to reduce the gap between the individual developers knowledge and what is needed. For instance, agents can minimize the process adoption thresholds so that the complexity of a process can become transparent to the developers and thus be perceived as lightweight. The individual developer will focus on the problem solving and creative part, letting the agents do the work that can be guided by formalized knowledge. We will discuss the process of formalizing knowledge as rules, how these rules will trigger in a given context and how the agents can propose resolutions. Examples will be used to demonstrate the feasibility of agents in software development.

Audience: System analysts, project leaders, software developers, people interested in methodologies, process development and software development tools.

Required experience: Some experience of software development, and of UML.

T03: Concurrent Object–Oriented Design in Java

Presenters: Doug Lea, David Holmes

Duration: full day

Level: Intermediate

Day: Monday (June 18)

This tutorial starts off with a careful presentation of Java language and library support for concurrent programming. It then proceeds with overviews of the design and implementation of styles and patterns that commonly appear in concurrent programs and systems. These include layered and transactional synchronization, monitors and conditions, optimistic synchronization, lightweight tasks, IO, message, and Event based frameworks, Futures, and computationally intensive parallel programming frameworks. Presentations include high-level design guidance, concrete code examples, as well as information about how to learn more about and experiment with associated packages and techniques.

Audience: Developers, programmers, students

Required experience: Familiarity with Java

Presenter's profile: Doug Lea is a professor of Computer Science at the State University of New York at Oswego. He is author of the book "Concurrent Programming in Java", and co-author of the text "Object-Oriented System Development". David Holmes is a Senior Research Scientist at the Cooperative ResearchCentre for Enterprise Distributed Systems Technology (DSTC Pty Ltd), in Brisbane, Australia. He completed his Ph.D in the area of synchronization within object-oriented system and has been involved in concurrent programming for a number of years. He is a co-author of the third edition of the Java Series book "The Java Programming Language".

T04: Design of Software Architectures

Presenter: Jan Bosch

Duration: half day

Level: Intermediate

Day: Monday (June 18) a.m.

One can identify an increasing identification of the importance of explicit design of software architectures. This tutorial presents a method for architectural design explicitly focussing on assessment of and transformation for quality attributes. Assessment of software architectures using scenario-, simulation-, metrics- and experience-based assessment approaches is discussed.

Transformation of software architectures is using architectural styles, architectural patterns, design patterns and by converting quality requirements to functionality. The design method is extensively illustrated by examples and experiences from numerous industrial cases. The tutorial is based on a recently published book Design and Use of Software Architectures - Adopting and Evolving a Product-Line Approach authored by the tutorial presenter and published by Addison-Wesley.

Audience: Software engineers and technical managers considering the introduction of explicit software architecture design in their organization. Researchers interested in the experiences collected by the tutorial presenter.

Required experience: It is assumed that the participant has some experience with industrial software development.

Presenter's profile: Prof. dr. ir. Jan Bosch is a professor of software engineering at the University of Groningen. His research activities include software architecture design, software product lines, object-oriented frameworks and component-oriented programming. He is the author of a book "Design and Use of Software Architectures: Adopting and Evolving a Product Line Approach" published by Pearson Education (Addison-Wesley & ACM Press).

T05: Enterprise JavaBean Architecture and Design Issues; Avoiding JavaBean Soup

Presenter: James White

Duration: half day

Level: Intermediate

Day: Monday (June 18) p.m.

Enterprise JavaBeans (EJB) have become a staple in distributed object and component architectures. However, like most technologies, EJB provides technology tradeoffs that must be weighted by the architect contemplating its use. Furthermore, like most technologies, EJB cannot save a poor system design.

Specifically, this tutorial will be separated into two general sessions. The first half of the tutorial will cover the larger architectural decisions surrounding EJB like decisions about whether EJB technology is right for a project and selecting an appropriate EJB server.

The second half of the tutorial will cover issues in bean design. These "micro" issues include: when and where to use entity versus session beans, when and where to use container managed versus bean managed persistence for entity beans, the granularity (fine or course) of EJBs, and the impact of the underlying database on bean design.

Audience: This tutorial is intended for software architects, analyst and developers examining or working with Enterprise JavaBean technology.

Required experience: Familiarity with EJB terminology and technology is helpful.

Presenter's profile: James White is co-founder and president of Catapult Technologies, Inc., an object-oriented/component consulting firm in the Twin Cities. He has over twelve years of software development experience and holds a M.S. in Computer Science from the University of Minnesota.

T06: Generative Programming: Methods, Techniques, and Applications

Presenters: Krzysztof Czarnecki, Ulrich W. Eisenecker

Duration: full day

Level: Intermediate

Day: Monday (June 18)

There are exciting new developments in Software Engineering: New "post-object-oriented" programming paradigms such as Aspect-Oriented Programming and Intentional Programming are emerging, and the recent work in Software Product-Line Engineering integrates classic Domain Engineering methods with object-oriented and component-based programming. Generative Programming builds upon these new developments to support the design and implementation of generative models of system families, which allow a specification-based generation of family members from reusable components. In this tutorial, the authors of the book "Generative Programming: Methods, Tools, and Techniques" (Addison-Wesley, 2000) will present the basic motivation for the shift to software system family engineering, an overview of the process of generative programming and its work products, generative programming techniques in C++ and Java, and a complete case study demonstrating all the steps from analysis to an implementation.

Audience: This tutorial is aimed at researchers and practitioners interested in cutting-edge approaches to achieve reusability and adaptability.

Required experience: Programming skills in OOP languages, common understanding of parametric polymorphism

Presenter's profile: Krzysztof Czarnecki is a researcher and consultant with the Software Technology Lab at Daimler Chrysler Research in Ulm. Ulrich Eisenecker is a professor of computer science at the University of Applied Sciences Kaiserslautern at Zweibrücken and an industry consultant.

T07: Adapting XP to Complex Application Domains

Presenters: Martin Lippert, Stefan Roock

Duration: half day

Level: Intermediate

Day: Monday (June 18) a.m.

XP is still a hot topic. It looks as if XP worked well for small software projects in not too complex application domains. In many of these projects the rather simple requirements engineering of XP (customers write story cards) was suitable. But today we are more often faced with complex application domains where the classical XP techniques will not suffice. The tutorial shows how to adapt extreme programming to complex application domains and for demanding development tasks. We focus mostly on the requirements engineering part and show how to enhance XP with interviews, scenarios and system visions. We also integrate the management perspective into the "planning game" reconciling this perspective with users' needs. We propose a set of best-practice methods, we have used in a number of industrial high-risk projects for different complex application domains.

Audience: Project leaders, software architects and experienced software developers who are interested in XP but sceptical using XP for their projects.

Required experience: The participants should have experience with object-oriented analysis and design. Extreme programming experiences are not presumed.

Presenter's profile: Martin Lippert and Stefan Roock are research assistants at the University of Hamburg and professional software architects and consultants at APCON WorkplaceSolutions. They are coaches for professional XP projects and gained experience in successful XP projects over the past years.

T08: Object–Oriented Frameworks

Presenter: Greg Butler

Duration: half day

Level: Intermediate

Day: Monday (June 18) p.m.

Frameworks offer a concrete realization of a product line. A framework is an architecture, plus an implementation, plus documentation that capture the intended use of the framework for building applications. A framework provides a highly effective mechanism for software reuse within an application domain. The framework captures the features that are common across the product line. In return for relinquishing some design authority, the developer can build a new application faster by hooking to the framework just the code that is unique to the new application. The tutorial presents methodologies for the development, application, and evolution of object-oriented frameworks. Concepts and techniques behind modelling and implementation of the commonality and variability within a domain are presented.
The framework maturity life cycle ranges from white-box frameworks, through composition of component-based systems, to generative techniques using domain-specific languages. We draw on our Know-It-All framework of database systems to provide case study material.

Audience: The tutorial is aimed at practitioners and researchers with experience in object-oriented design, and interest in reuse and product lines.

Required experience: The tutorial assumes the basic concepts: objects, polymorphism, delegation; and experience in object-oriented design. Knowledge of design patterns is beneficial.

Presenter's profile: Dr Butler researches methodologies for framework evolution, and is developing a framework for databases with applications to bioinformatics. He has written over 50 technical papers, and has consulted on object-oriented design, object-oriented technology, database technology, and large-scale software architecture.

T09: Programming with XML

Presenter: Peter Bunemann

Duration: half day

Level: Intermediate

Day: Monday (June 18) a.m.

XML is widely believed to be the future medium for data exchange on the Web, but how will we program with it? In asking this question we have to remember that large XML documents will be stored in databases, and that our programming language interfaces will have to provide more than a serial scan of the document. Some form of query language will be essential. In this tutorial I shall cover three topics:

- XML query languages. These include XSL, Lorel, XML-QL and XQL. The more sophisticated of these query languages are based on ideas that were developed for semistructured data. They have certain common features but differ greatly in their expressive power.
- Programming interfaces. A brief review of the Document Object model and how it is exploited in current systems.
- Programming languages and more sophisticated APIs. The systems mentioned above are essentially untyped. They ignore Document Type Declarations (DTDs) which are part of the XML standard and bear some resemblance types. Can we use them to provide new languages and more robust APIs? Prototypes such as YAT and Xduce offer new approaches to subtyping and also indicate shortcomings in DTDs.

Audience: People interested in XML and programming with it.

Required experience: Knowledge of XML Database query languages (helpful but not essential) Types and subtypes Basic math for CS (e.g. automata theory)

T10: Squeak: An Open Source Smalltalk for the 21st Century!

Presenter: Andrew P. Black

Duration: half day

Level: Intermediate

Day: Monday (June 18) p.m.

Squeak is an open source Smalltalk system that runs on many platforms (including Windows, MacOS, Linux, and many PDAs). It is fun to use, has a large, talented and enthusiastic user community, and is constantly improving. Because all the source code is written in Squeak itself and is freely available, anything can be changed to suit the needs of the programmer. Squeak is ideal as an experimental, exploratory environment, as well as for teaching.

This tutorial is intended for those familiar with object-oriented concepts and design, and who are keen to explore the richness of a 21st Century Smalltalk. The tutorial will be "hands-on"; participants are encouraged to bring their own laptop computers loaded with Squeak. We will cover the essential aspects of conventional Smalltalk, such as the programming environment, debugging, and testing, and will emphasize Squeak's innovations, such as the Morphic graphic model, 3D, book-morphs, scripting, and Sound.

Audience: Those seeking a productive programming environment that fulfills the promise of Object-Orientation, or wishing to explore UI-design, Multimedia and Graphics.

Required experience: Participants should know about O-O concepts and O-O design. They are likely to have programmed in another O-O language.

Presenter's profile: Andrew Black is a Professor at the Oregon Graduate Institute of Science and Technology, located just outside Portland, Oregon, USA, where he has taught MS and PhD students Object-Oriented Programming using Smalltalk and other languages since 1994. He has been involved in OO language and systems research since 1981.

T11: Pattern Mastery through Pattern Writing

Presenter: James O. Coplien, Christa Schwanninger

Duration: 2 days

Level: Introductory

Day: Monday-Tuesday (June 18-19)

Software patterns are a literary form, designed to communicate expert knowledge about system construction. The most useful patterns are the ones that address structural problems, and which are carefully written to be readable.

This course teaches pattern-writing skills in the context of the broader goals and values of the pattern community. The course mixes a small amount of lecture with exercises and small workshops. Attendees do a comparative analysis of published patterns. On the second day, participants work in small teams to write patterns, and participate as an author in a writer's workshop.

Audience: People who wish to start or enhance pattern programs at their companies and academic institutions, as well as those justwishing to know a bit more about the foundations of patterns. The focus will be on software but we can accommodate people from a wide variety of intellectual and artistic endeavors.

Required experience: None

Presenter's profile: Jim Coplien is a Distinguished Member of Technical Staff at Bell Laboratories in Naperville, Illinois. He is the author of the "Software Patterns" management briefing and co-editor of several pattern books. Christa Schwanninger is a software engineer at Siemens Corporate Technology in Munich where she works on distributed applications. She has done pioneering work in patterns for compiler construction and language design.

T12: Aspect–Oriented Programming with AspectJ™

Presenters: Erik Hilsdale, Gregor Kiczales

Duration: full day

Level: Intermediate

Day: Tuesday (June 19)

AspectJ is a seamless aspect-oriented extension to Java. It can be used to cleanly modularize the crosscutting structure of concerns such as exception handling, multi-object protocols, synchronization, performance optimizations, and resource sharing. When implemented in a non-aspect-oriented fashion, the code for these concerns typically becomes spread out across entire programs. AspectJ controls such code-tangling and makes the underlying concerns more apparent, making programs easier to develop and maintain.

This tutorial will show how to use AspectJ to implement crosscutting concerns in a concise, modular way. We will use numerous examples to develop participant's understanding of AspectJ andaspect-oriented programming. We will also demonstrate AspectJ's integration with IDEs such as JBuilder 3.5 and Forte4J, and emacs.

Audience: Participants will learn how to use AspectJ, as a natural extension of their existing Java development, to improve the modularity, reusability and maintainability of their systems.

Required experience: Attendees should have experience doing object-oriented design and implementation, and should be able to read Java code. No prior experience with aspect-oriented programming or AspectJ is required.

Presenter's profile: Erik Hilsdale is a member of the research staff at Xerox's Palo Alto Research Center. As a member of the AspectJ project team, he focuses on language design, pedagogy and compiler implemetation. Gregor Kiczales is Professor of Computer Science and Xerox/Sierra Systems/NSERC Chair of Software Design at the University of British Columbia. He is also a Principal Scientist at the Xerox Palo Alto Research Center, where he leads the group that has developed aspect-oriented programming and AspectJ.

T13: Patterns at Work

Presenter: Frank Buschmann

Duration: half day

Level: Advanced

Day: Tuesday (June 19)a.m.

In this tutorial we present in detail a part of a concrete real-world system and how it is designed with patterns: the representation of physical storage in a warehouse management system as well as the client interface to this subsystem. Step by step we will re-play the process of the system's construction. We discuss the design problems that occur, present the patterns that could help solving these problems, discuss design alternatives and show how we actually applied the patterns we selected. By this we will see how the design of the system slowly grows and evolves towards the final architecture. we will also see and discuss how patterns are applied in practise and how they help building high-quality software with predictible properties.The tutorial concludes with a summary of our experiences from several projects in which we applied patterns: what worked, what could be improved, and what did we learn.

Audience: Software-Engineers (professionals, students) who are interested in using patterns for designing systems.

Required experience: Sound knowledge in Object Technology, Knowledge of UML Notation Knowledge of the pattern concept.

Presenter's profile: Frank Buschmann is software engineer at Siemens Corporate Technology in Munich, Germany. His interests include Object Technology, Frameworks and Patterns. Frank has been involved in many software development projects. He is leading Siemens' pattern research activities. Frank is co-author of "Pattern-Oriented Software Architecture -- A System of Patterns" and "Pattern-Oriented Software Architecture -- Patterns for Concurrent and Networked Objects"

T14: Advanced Visual Modelling: Beyond UML

Presenters: Joseph (Yossi) Gil, John Howse, Stuart Kent

Duration: half day

Level: Intermediate

Day: Tuesday (June 19) p.m.

With the adoption of UML by the OMG and industry as the linguae-francae of visual systems modelling, one begins to ponder what will come next in this field? This tutorial brings a vision for With the adoption of UML by the OMG and industry as the linguae-francae of visual systems modelling, one begins to ponder what will come next in this field? This tutorial brings a vision for visual modelling beyond UML. We present several new notations, with quickly increasing adoption by industry, for the specification of complex systems in an intuitive visual, yet precise manner: Spider diagrams considerably extend Venn-diagrams to the specification of OO-systems. Familiar OO-concepts are translated to set theoretical terms: class into set of objects, inheritance corresponding to subset, and even Harel's statecharts interpreted as the set of objects in that state. Constraint diagrams enhance the arrow notation to describe static system invariants which cannot be described by UML class-object diagram. Reasoning rules are developed for the notation and strong completeness results are given. Finally, 3D-diagrams show how the third dimension and VRML modelling can be used for a conceptual modelling of dynamic system behaviour.

Audience: technology seekers, modellers in industry, researchers and students interested in design, modelling, and visual languages.

Required experience: Basic OO programming experience.

Presenter's profile: Dr Gil served as the head of the software and systems development laboratory at the Technion. Research interests include OO applications of theoretical computer science. Chair of TOOLS USA'98 and on the programme committees for numerous conferences including past & present OOPSLAs and ECOOPs. An experienced university teacher and trainer to industry.

T15: Interconnecting Objects via Contracts

Presenters: José Luiz Fiadeiro, Luís Filipe Andrade

Duration: half day

Level: Intermediate

Day: Tuesday (June 19) a.m.

Interactions between objects are too often coded in the way messages are exchanged and methods are called, making it difficult to understand and evolve the way components are interconnected without a deeper analysis of the way objects are designed. Yet, most businesses require high levels of reconfigurability of existing services, which suggests that an explicit model of the coordination that is required between system components should be given immediately during domain modelling.

Capitalizing on work in Software Architectures and Coordination Languages, and our experience in the banking industry, this tutorial puts forward the concept of contracts for providing explicit representations of interconnections between objects.

Audience: Business/Domain Analysts, Software Architects, Programmers, Researchers.

Required experience: Familiarity with object-oriented conceptual modeling.

Presenter's profile: José Fiadeiro is Professor at Lisbon University. Luís Andrade is partner of Oblog Software SA. They are partners of a new company dedicated to Architectural IT solutions already responsible for the technical architecture of Banco Espirito Santo in Portugal.

T16: Advanced Software Composition: Obstacles & Approaches

Presenters: Lodewijk Bergmans, Mehmet Aksit

Duration: one day (half day)

Level: Intermediate/Advanced

Day: Tuesday (June 19) p.m.

The object-oriented paradigm has been successful because of its good modularity characteristics, which supports the separation of concerns from analysis to implementation phases. Composability problems may be experienced when constructing new objects from existing ones, for example when objects need to evolve due to new or changing requirements. This experience has triggered researchers in the past years to come up with enhancements of the OO model in an attempt to solve the composability problems. The most well-known examples in this area are: AOP/AspectJ, Subject-Oriented Programming & HyperJ, Adaptive Programming and Composition Filters. [ECOOP has featured many workshops where the issues involved and solution proposals were discussed (ranging from the "Composability in OO" workshop in 1996 to the "Aspects and Dimensions of Concerns" workshop in 2000). Workshop participation in ECOOP/ICSE/OOPSLA workshop in 2000 illustrates the growing interest in this area.] This tutorial tries to explain this area starting from the problems of the OO model and design patterns. It presents an analysis of the so-called composition anomaly. It discusses the important issues that must be addressed during the design of objects/concerns so that they are composable at all. Then an overview of the current state-of-the-art is given, based upon a description of the design space for advanced composition approaches.

Audience: This tutorial is intended for software professionals and researchers who want to gain an understanding of the origins, issues and approaches for advanced software composition models. The tutorial focuses on understanding the real issues involved and comparing the state-of-the art based on this understanding, rather than explaining the specifics of the various approaches.

Required experience: Extensive experience with OO modelling, design and programming is required, common design patterns (Gamma et. al.) should have no secrets.

Presenter's profile: Lodewijk Bergmans and Mehmet Aksit have both been working on software composition for over a decade. Their work has included the analysis of inheritance anomaly for synchronization and real-time specifications (composability problems) and the composition filters approach to solve such issues. They have been involved in the organization of most workshops in this area. Both are experienced teachers who have together given over 100 professional (open/international/in-company) courses. Mehmet Aksit has given several tutorials during ECOOP and OOPSLA in the past.

T17: Introduction to Java–based Mobile Agent Programming Systems

Presenter: Anand Tripathi

Duration: half day

Level: Intermediate

Day: Tuesday (June 19) a.m.

In recent years, there has been a surge of interest in autonomous mobile objects and mobile agent systems. Several mobile agent programming platforms have been developed by the research community to support mobile programs in distributed systems. Most of today's commonly used mobile agent platforms are designed and implemented using Java and its security architecture. Agents are implemented as transportable Java objects. This tutorial presents an overview of the various design issues in a mobile agent programming system. These include migration support, global naming, inter-agent communication, protection of host resources, protection of agent state, agent authorization, etc. This tutorial surveys a number of Java-based agent programming systems with respect to these design issues. These include Aglets, Ajanta,Concordia, D'Agents, Mole, and Voyager. Finally, a number of potential applications of the mobile agent paradigm are discussed and conclusions are drawn based on the reported experiences.

Audience: Researchers, educators, systems programmers and application developers who want to gain a better understanding of the benefits and limitations of the mobile agent paradigm.

Required experience: Some familiarity with basic concepts in distributed computing with objects; some familiarity with Java's programming model.

Presenter's profile: Anand Tripathi is an Associate Professor at the University of Minnesota. He received Ph.D. in Electrical Engineering from the University of Texas at Austin in 1980. His current research is in secure distributed computing using mobile agents. His group has developed a mobile agent system called Ajanta.

T18: Developing Applications for the J2ME™ Mobile Information Device Profile

Presenters: Aldo Eisma, Michael Brung

Duration: half day

Level: Intermediate

Day: Tuesday (June 19) p.m.

The Mobile Information Device Profile (MIDp) defines a set of APIs for writing portable Java™ applications for the smallest mobile devices, like for example mobile phones, that fit the constraints of the Connected Limited Device Configuration (CLDC) of the Java 2 Micro Edition platform. The MIDp and CLDC specifications are the result of cooperative work under the Java Community Process. A similar profile for somewhat larger devices, like Personal Digital Assistants, is currently being specified.

In this tutorial we will introduce these profiles and configurations, and discuss our experience with implementing the run-times for these profiles for IBM VisualAge Micro Edition, and developing a portable conference scheduling application using the MIDp. We will demonstrate and look at the internals of the libraries and the application to illustrate the possibilities and limitations of these new profiles.

Audience: Programmers and architects of mobile embedded systems who want to learn how Java can help them, and Java developers who want to find out how to develop a mobile embedded Java application.

Required experience: Basic knowledge of developing applications and libraries in the Java programming language, and familiarity with handheld devices.

Presenter's profile: Aldo Eisma is the lead developer for the SmartLinker Java application packager, a component of IBM VisualAge Micro Edition developed by Object Technology International, at Amstelveen, The Netherlands. He has also contributed to the development of the PalmOS run-time environment for IBM VisualAge Micro Edition. Michael Brung is Software Developer for Object Technology International. He is involved in the development of embedded Java applications for PalmOS, and is responsible for the release and the maintenance of the PalmOS run-time environment for IBM VisualAge Micro Edition.

T19: Component Technologies for the Middle−Tier

Presenter: Michael Stal

Duration: half day

Level: Advanced

Day: Tuesday (June 19) a.m.

Almost all distributed object-oriented systems are partitioned into multiple tiers where the presentation tier contains the user interface and the database tier connects the application to databases and other legacy applications. In the middle between them the business logic is provided. Needless to say that the middle tier is the most important part of this architecture. The middle tier must be easy to develop, easy to manage, and easy to change and adapt. Thus, building the middle tier as a monolith is not an appropriate approach. The right approach is to use a component-based approach to separate the concerns within the middle tier. It is the goal of the tutorial to introduce the middle-tier component technology in general as well as the different solutions available today.

Audience: People interested how to build component-based middle-tier components. Developers interested in COM+, CORBA 3 and Enterprise JavaBeans.

Required experience: Attendees should be familiar with object-oriented technologies. Knowledge of components and distribution technologies is recommended but not required.

Presenter's profile: Michael Stal works as a Senior Principal Engineer for Siemens Corporate Technology where he is head of the Middleware & Application Integration Team. Michael is co-author of the "Pattern-Oriented Software Architecture" book series and editor-in-chief of the German Java Spektrum magazin.

T20: Real Time System Modelling with UML: Current Status and Some Prospects

Presenter: François Terrier, Sébastien Gerard

Duration: full day

Level: Intermediate/Advanced

Day: Tuesday (June 19)

Engineers are more and more faced to the hard problem of developing more sophisticated real-time systems while time to market and cost constraints are constantly increasing. Current work performed at the OMG on UML standard evolutions to better integrate real-time issues shows both that there is a strong interest on the subject and that current proposals are neither completely satisfying, neither completely compatible. In particular, some available methods (namely ARTiSAN, RT-UML, UML/SDL and UML-RT) provide a good support for parallelism modelling of an application, but they are often poor to express quantitative real-time features (such as deadlines, periods, priorities...). Moreover they are always based on proprietary solutions not obviously fully compliant with the UML standard or/and OO view. So, two important challenges are still targeted: first to define how to support high level real time specification concepts in UML; second, to unify them within these proposed by existing tools. This tutorial will study the current status of the main existing methods and tools and propose some directions to manage the remaining problems.

Audience: Engineers developing embedded systems, researcher and teachers concerned by software engineering practices for real time systems.

Required experience: Knowledge of the real time system domain and preferably general knowledge on UML.

Presenter's profile: Doctor in Electronic, François Terrier is Professor at the National Institute of the Nuclear Sciences and Technologies. He works at the CEA - French Atomic Energy Agency for over 10 years and is leader of the Software for Process Safety Laboratory. Sébastien Gerard is Doctor in computer science and graduated from the Superior School of Mechanics and Aeronautics. His work is centred on modelling embedded systems for automotive industry in collaboration with the French car maker, PSA (Peugeot-Citroen Automobile).

T21: Security and Object Oriented Programming

Presenter: Dieter Gollmann

Duration: half day

Level: Intermediate

Day: Tuesday (June19) p.m.

This tutorial will introduce the conceptual issues currently faced in information security and move on to examine the relationship between security and OOP, investigating examples like the CORBA security architecture.

Audience: This tutorial is indeed for researchers and practitioners who want to get a more thorough understanding of general security issues so that they can better assess the contributions OOP can make to security.

Required experience: None, other than interest in security issues; the less preconceived ideas about security the better.

Presenter's profile: Before joining Microsoft Research, I was the first Course Director of the MSc in Information at Royal Holloway, University of London. I have published a textbook on 'Computer Security' and am a co-editor-of-chief of Springer's International Journal of Information Security.

Workshops

Each workshop lasts one day (except for W1 and W10). Workshops take place on Monday and Tuesday (except for W10, which takes place on Sunday and Monday) in the Northern Block of the ELTE University. They will be held from 9:00 to 17:30.

Note that the ECOOP workshops are usually open only to persons who are invited or who have submitted a position paper. It is the responsibility of the registrant to check whether he or she is in the appropriate situation.

Workshop timetable

	Title	*Mon*	*Tue*
W1	11th Workshop for PhD Students in Object-Oriented Systems *Dr. Charles Suscheck, Gilles Ardourel, Jose Luis Herrero Agustin, Michael Haupt, Rainer Ruggaber*	☒	☒
W2	2nd Workshop on Quality of Service in Distributed Object Systems *Christian Becker, John Zinky*		☒
W3	Distributed Multimedia Object/Component Systems *L. Böszörményi, C. Becker, C. Stary, H. Kosch*	☒	
W4	Formal Techniques for Java Programs *Arnd Poetzsch-Heffter, Erik Poll, Gary Leavens, Peter Müller, Susan Eisenbach*	☒	
W5	Automating Object-Oriented Software Development Methods *Bedir Tekinerdogan, Pim van der Broek, Motoshi Saeki, Pavel Hruby, Gerson Sunyé*		☒
W6	Adaptive Object-Models and Metamodeling Techniques *Ali Arsanjani, Joseph W. Yoder, Nicolas Revault*	☒	
W7	Specification, Implementation and Validation of Object-oriented Embedded Systems *Bran Selic, Francois Terrier, Paul Pettersson, Sebastien Gerard, Udo Brockmeyer, Wang Yi, Werner Damm*		☒

	Title	*Mon*	*Tue*
W8	Feature Interaction in Composed Systems *Andreas Speck, Elke Pulvermueller, James O. Coplien, Maja D'Hondt, Wolfgang De Meuter*	☒	
W9	6th International Workshop on Component-Oriented Programming *Clemens Szyperski, Jan Bosch, Wolfgang Weck*		☒
W10	Advanced Separation of Concerns (Sun: ☒) *Johan Brichau, Lodewijk Bergmans, Maurice Glandrup, Siobhan Clarke*	☒	
W11	Multiparadigm Programming with OO Languages *Marion Kei Davis, Yannis Smaragdakis, Jörg Striegnitz*	☒	
W12	Generative Programming *Barbara Barth, Greg Butler, Krzysztof Czarnecki, Ulrich Eisenecker*		☒
W13	4th Workshop on Object-Oriented Architectural Evolution *Galal Hassan Galal, Tom Mens*	☒	
W14	Scientific Computing Environment: CORBA Components for the GRID *László Varga, Zsolt Kovács*	☒	
W15	Development of Robust and High Confidence Agent Applications *Ciarán Bryce, Alexander Romanovsky, Anand Tripathi*	☒	
W16	Mining Pedagogical Patterns *Joseph (Joe) Bergin, Jutta Eckstein*		☒
W17	The Next 700 Distributed Object Systems *Andrew Black, Anne-Marie Kermarrec, Doug Lea, Eric Jul, Jochen Liedtke, Rachid Guerraoui, Salah Sadou*		☒
W18	5th ECOOP Workshop on Quantitative Approaches in Object-Oriented Software Engineering *Brian Henderson-Sellers, Fernando Brito e Abreu, Geert Poels, Houari A. Sahraoui, Mario Piattini*	☒	

	Title	*Mon*	*Tue*
W19	Mechanisms for Enterprise Integration: From Objects to Ontologies *Mark Lycett, Sergio de Cesare*		☒
W20	Engineering Complex OO Systems for e-Commerce *Simon Dobson, Siobhán Clarke, Vinny Cahill*		☒
W21	Workshop on Object-Oriented Business Solutions *Antonio Ruiz, Jesús D. García-Consuegra, Jörg Mühlbacher, Rafael Corchuelo*	☒	
W22	4th ECOOP Workshop on OO and Operating Systems *Darío Álvarez Gutierrez, Francisco Ballesteros, Paniti Netinant*		☒

Workshop reader

This year we will again propose a Workshop Reader, edited by Springer-Verlag and containing reports on the workshops.

W01: 11th Workshop for PhD Students in Object−Oriented Systems

Organizers: Gilles Ardourel, Michael Haupt, Jose Luis Herrero Agustin, Rainer Ruggaber, Dr. Charles Suscheck

Days: Monday, 18 June, Tuesday, 19 June

This workshop is slightly unusual, because the participants are PhD students and the topics are derived from the areas of interest of the participants. The workshop is divided into plenary sessions with a number of pre-screened presentations, and "discussion" sessions, holding small subgroups composed of PhD students working on similar topics. For each participant, this is an opportunity to present his/her research to a knowledgeable audience working in a similar context, and to share his/her ideas on hot-topics or new trends. In this way, the participants may receive insightful comments on their research, learn about related work, and initiate future research collaborations.
The participants are divided into three categories. First, it is possible to submit a (3-8 page) extended abstract on a specific topic, and give a 30 minutes presentation at the workshop. Second, a PhD student may submit a one-page abstract of their PhD project, and give a 15 minutes talk. Finally, last-minute participants may contribute with a short and informal oral presentation of their research.
The program committee is essentially composed of PhD students and young researchers with a strong background on some area of object-orientation. The review process is not designed to select the few very best papers, but to ensure that every participant is able to present some relevant material, and is well prepared.

W02: 2nd Workshop on Quality of Service in Distributed Object Systems

Organizers: Christian Becker, John Zinky

Day: Tuesday, 19 June

The suitability of the object model for the modeling and implementation of distributed systems has lead to middleware platforms, such as e.g. CORBA, DCOM, Java/RMI, and recently .Net and SOAP. Originally, these middleware systems aim at distribution transparency for application programmers.

However, distributed systems are exposed to system issues, like dynamic performance changes or partial errors, that prevent a complete distribution transparency. Quality of Service (QoS) management addresses these issues. The goal is to add QoS management to the interactions between clients and services.

Support for QoS management in distributed object systems is a hot topic of current research which poses a number of open questions: How is QoS integrated with the object model that emphasizes encapsulation and information hiding? Can one build generic support frameworks for multiple QoS categories, in contrast to specialized, single category systems, such as TAO, Electra, Eternal, DOORS among others. Can QoS be viewed as an aspect in the sense of Aspect Oriented Programming (AOP) or are other classifications more appropriate?

Based on nowadays object-oriented middleware platforms component based systems emerge. The necessity for QoS provision upholds for components as well. The integration of QoS specifications in component models as well as the mapping to underlying QoS provision of object systems is a challenge of the near future.

The proposed ECOOP-workshop will discuss the open questions and will summarize the state of the art in the field. It is expected that the workshop will stimulate discussions about how next generation QoS management facilities can be built into object infrastructures. We expect the workshop to attract researchers from several groups which have working prototypes for this infrastructure. Reports on prototype systems are welcome as well as theoretical aspects, modeling techniques, and application scenarios.

W03: Distributed Multimedia Object/Component Systems

Organizers: L. Böszörményi, C. Stary, H. Kosch, C. Becker

Day: Monday, 18 June

The design and implementation of distributed multimedia systems (especially multimedia infrastructure) show a high divergence among proprietary systems - multimedia standards being the only glue keeping them working together. If distributed multimedia is to become a key technology for the next decade then it must adopt object and component-based philosophies. Nowadays, multimedia systems are implemented partly in an ancient, monolithic style, which is acceptable for the childhood of a new technology, but which will not work in its mature phase.

Much effort has been invested in developing frameworks to support the client-side of multimedia applications (such as Java Media Framework). Modern middleware approaches also incorporate both component-oriented and quality-of-service aspects (such as Corba 3.0 and Enterprise Java Beans). Object-relational databases provide progressively better support for storing and streaming multimedia data (such as Oracle Video Server). Multimedia standards, more and more, are covering the structuring of multimedia data (such as video-objects in MPEG-4) and are adding semantic information (such as MPEG-7). Nevertheless, relatively little attention has been paid so far to the software technology of entire, large-scale distributed multimedia systems (incl. applications and infrastructure). It is urgently needed to build fully connected bridges between the three still more or less independent areas: distributed, multimedia and object-oriented systems. These bridges might either stem from modeling approaches for designing or from implementation techniques for distributed multimedia systems.

The aim of the workshop is to explore the potential of object and/or component technology in building complex distributed multimedia systems.

W04: Formal Techniques for Java Programs

Organizers: Susan Eisenbach, Gary Leavens, Peter Müller, Arnd Poetzsch-Heffter, Erik Poll

Day: Monday, 18 June

Formal techniques can help to analyze programs, to precisely describe program behavior, and to verify program properties. Applying such techniques to object-oriented technology is especially interesting because:
The OO-paradigm forms the basis for the software component industry with their need for certification techniques.
It is widely used for distributed and network programming.
The potential for reuse in OO-programming carries over to reusing specifications and proofs.
Java is a good platform to bridge the gap between formal techniques and practical program development. It plays an important role in these areas and is on the way to becoming a de facto standard because of its reasonably clear semantics and its standardized library. However, the language Java contains novel language features, which are not fully understood yet. Furthermore, Java supports a novel paradigm for program deployment, and improves interactivity, portability and manageability. However, this paradigm also opens new possibilities for abuse and causes concern about security.
Thus, work on formal techniques and tools for Java programming and formal underpinnings of Java complement each other. This workshop aims to bring together people working in these areas.

W05: Automating Object–Oriented Software Development Methods

Organizers: Bedir Tekinerdogan, Pim van den Broek, Motoshi Saeki, Pavel Hruby, Gerson Sunyé

Day: Tuesday, 19 June

Numerous object-oriented software development methods exist in the literature. Most popular methods have a general character, but some methods, like real-time system design, are targeted at specific application domains. Some methods are specifically defined for a given phase in the life cycle of software development, such as requirement analysis or domain analysis. It is generally accepted that these methods are useful for developing high-quality software.

Most methods methods usually include a number of different heuristic rules, which are needed to produce or refine different artifacts. Moreover, the rules are structured in different ways, leading to different software development processes. Although useful, applying methods is a complex issue, and does not necessarily lead to effective and efficient software development. Automated support for object-oriented methods will decrease this complexity, increase reusability, and provide better support for adaptability, customizability and continuous improvement. Unfortunately, apart from the many environments with diagram editors and visualization tools, existing object-oriented methods are only basically described in separate handbooks and manuals. Complete and integrated tools, which support the entire life cycle, are not yet present in practice.

This workshop aims to identify the fundamental problems of automating methods and to explore the mechanisms for constructing case tools that provide full support for methods.

W06: Adaptive Object–Models and Metamodeling Techniques

Organizers: Nicolas Revault, Joseph W. Yoder, Ali Arsanjani

Day: Monday, 18 June

A system with an Adaptive Object-Model (AOM) has an explicit object model that it interprets at runtime. If you change the object model, the system changes its behavior. For example, a lot of workflow systems have an Adaptive Object-Model. Objects have states and respond to events by changing state. The Adaptive Object-Model defines the objects, their states, the events, and the conditions under which an object changes state.

There are various techniques that share common features with AOM's. Especially, those that try also to capture business rules and build domain specific languages, namely ? Grammar-oriented Object Design (applied in the three major areas of configurable workflow, tier-to-tier mapping and object graph traversal) or ? Meta-CASE tools and environments approaches, à la MetaEdit+ or à la MétaGen (applied in various fields of information system modeling: telecom, finance, medicine, etc.). There are other techniques which also describe ways to build systems that change behavior at runtime, namely ? Reflection at the language level (mostly applied to programming language design). What is actually common to those various techniques is that they are leading to, or are driven by, metamodeling principles and implementation using OO languages.

This workshop will focus on identifying, cataloging and comparing these techniques one towards another. We will also try to establish the conditions of use of these techniques, look at where they meet or overlap, and hopefully set some cross-fertilization ideas of benefit for each technique.

W07: Specification, Implementation and Validation of Object–Oriented Embedded Systems

Organizers: Sébastien Gerard, François Terrier, Bran Selic, Werner Damm, Udo Brockmeyer, Wang Yi, Paul Pettersson

Day: Tuesday, 19 June

This workshop aims to gather academics and industrial people to discuss the use of UML for the development of embedded real-time systems. Around UML technologies, we will try to tackle the three main part of a development cycle:

- Specification (or analysis) issues;
- Implementation issues;
- Validation, verification and testing issues.

W08: Feature Interaction in Composed Systems

Organizers: Elke Pulvermueller, Andreas Speck, James O. Coplien, Maja D'Hondt, Wolfgang De Meuter

Day: Monday, 18 June

The history of computer science has shown that decomposing software applications helps managing their complexity and facilitates reuse, but also bears challenging problems still unsolved, such as the assembly of the decomposed features when non-trivial feature interactions are involved. Examples of features include concerns or aspects, black box or white box components, and functional and non-functional requirements. Approaches such as object-oriented and component-based software development, as well as relatively new directions such as aspect-oriented programming, multi-dimensional separation of concerns and generative programming, all provide technical support for the definition and syntactical assembly of features, but fall short on the semantic level, for example in spotting meaningless or even faulty combinations. At previous ECOOPs, OOPSLAs and GCSEs dedicated events have been organised around the aforementioned technologies, where we experienced a growing awareness of this feature interaction problem. However, feature interaction is often merely dismissed as a secondary problem, percolating as an afterthought while other issues are being addressed. This workshop intends to be the first co-ordinated effort to address the general problem of feature interaction in composed systems separately from other issues.
All submissions compliant to a provided guideline will be selected from a review committee of well-known experts in this domain. A small set of submissions that raise important issues and are most likely to ignite discussions will be presented in the plenary session at the beginning of the workshop. The major part of the workshop will be spent on group work, ending in presentations of the group results. Collaborative work continues after the workshop, since each group will produce a small report which will be collected into the workshop report.

W09: 6th International Workshop on Component–Oriented Programming

Organizers: Jan Bosch,Clemens Szyperski, Wolfgang Weck

Day: Tuesday, 19 June

WCOP 2001 seeks position papers on the important field of component-oriented programming (COP). WCOP 2001 is the sixth event in a series of highly successful workshops, which took place in conjunction with every ECOOP since 1996.

COP has been described as the natural extension of object-oriented programming to the realm of independently extensible systems. Several important approaches have emerged over the recent years, including CORBA/CCM, COM/COM+, JavaBeans/EJB, and most recently .NET. After WCOP'96 focused on the fundamental terminology of COP, the subsequent workshops expanded into the many related facets of component software. WCOP 2001 shall emphasise the relationship between software architecture and component software. Two topics of particular interest are components vs. generators to address architectural variability and the component/connector distinction. In addition, submissions reporting on experience with component-oriented software systems in practice are strongly encouraged, where the emphasis is on interesting lessons learned, whether the actual project was a success or a failure.

To enable lively and productive discussions, the workshop will be limited to 25 participants. Depending on the submitted position papers, the workshop will be organized into three or four subsequent mini-sessions, each initiated by a presentation of two or three selected positions and followed by discussions. Instead of splitting the workshop into task forces, it is intended to provoke lively discussion by preparing lists of critical questions and some, perhaps provocative, statements (to be used on demand).

Position papers will be formally reviewed, each by at least two independent reviewers. As an incentive for submission of high quality statements, the best position statements will be combined with transcripts of workshop results and published.

W10: Advanced Separation of Concerns

Organizers: Lodewijk Bergmans, Maurice Glandrup, Johan Brichau, Siobhan Clarke

Day: Sunday, 17 June, Monday, 18 June

Recent approaches such as adaptive programming, aspect-oriented programming, composition filters, hyperspaces, role-modeling, subject-oriented programming and many others, as presented at previous ECOOP workshops, have enhanced object-oriented programming by providing separation of concerns along additional dimensions, beyond "objects". This is an exciting and active research area, with the potential to deliver far more flexible and effective separation of concerns.

A series of related workshops on "Composablity in OO", "Aspect-Oriented Programming" and "Aspects & Dimensions of Concerns" that have been held at ECOOP, as well as related workshops at ICSE and OOPSLA, indicate a fast growing interest in this area. Last year's ECOOP workshop titled "Aspects & Dimensions of Concerns" was considered very succesful by many people, not only because of the interest in the topic, but also since it managed to create a setting where people could really work together on example problems and requirements for this field. This year, we strive for a similar result, most likely by following a similar format.

The workshop will combine tigthly focused work in small groups with regular short plenary sessions. We require the participants to prepare the topic by (a) restricting the scope of the position papers and (b) setting up the groups and their topics before the workshop. The exact focus is yet to be determined, but we envision a followup on last years workshop results, such as the identification of the requirements and design space of solution approaches.

W11: Multiparadigm Programming with OO Languages

Organizers: Marion Kei Davis, Yannis Smaragdakis, Jörg Striegnitz

Day: Monday, 18 June

While OO has become ubiquitous for design, implementation, and even conceptualization, many practitioners recognize the need for other programming paradigms, according to problem domain. We seek answers to the question of how to address the need for other programming paradigms in the general context of OO languages.

Can OO programming languages effectively support other programming paradigms? The tentative answer seems to be affirmative, at least for some paradigms; for example, significant progress has been made for the case of functional programming in C++. Additionally, several efforts have been made to integrate support for other paradigms as a front-end for OO languages (the Pizza language, extending Java, is a prominent example).

This workshop seeks to bring together practitioners and researchers in this developing field to 'compare notes' on their work--that is, to describe techniques, idioms, methodologies, language extensions, software, or supporting theoretical work for expressing non-OO paradigms in OO languages. Work-in-progress descriptions are welcome, as are 'experience' papers if they present a lesson to be learned.

W12: Generative Programming

Organizers: Barbara Barth, Greg Butler, Krzysztof Czarnecki, Ulrich Eisenecker

Day: Tuesday, 19 June

The workshop aims to bring together practitioners, researchers, academics, and students to discuss the state-of-the-art of generative programming (GP), its relation to object-oriented programming and to other emerging approaches such as Aspect-Oriented Programming or Multidimensional Decomposition, and its role in software-engineering in general.
The goal is to share experience, consolidate successful techniques, analyze the relations between the various approaches, and identify open issues for future work.
Much of the industry focus has been on reusable components, but components still need to be assembled into concrete products. GP can help us to capture the configuration knowledge for a product line and use it to generate concrete family members. This step can be compared to the introduction of automated assembly lines in manufacturing.
The workshop will aim to foster discussion and interaction rather than presentations. Presentations will serve to introduce a case study, provoke discussion by presenting a controversial point of view, or introduce new points of view.
Our actual schedule, size, and format will depend on the number and quality of submissions, but in general we wish to promote discussion, and we wish to remain concrete.
Selection of Participants: Potential participants are asked to submit a two-page position paper detailing their experience with GP, their perspective on the relation of GP and other emerging approaches, and their planned contribution to the workshop. Based on the position papers, the organizers will invite a cross-section to participate. Dissemination of Results: A web-site will collect position papers, case studies, and presentations as well as a report on the discussions at the workshop.

W13: 4th Workshop on Object–Oriented Architectural Evolution

Organizers: Tom Mens, Galal Hassan Galal

Day: Monday, 18 June

This workshop is the fourth in a series of workshops in the area of software architecture and its evolution, which took place during ECOOP 1998, ECOOP 1999 and ECOOP 2000.
Previous workshops have proved very successful and stimulating, culminating in reports that contained novel and exciting views on what software architecture is, or should be, and how architectural issues may be approached from fresh perspectives. Past workshops also incorporated relevant experience reports and suggestions for future research in the area of evolving software architectures, especially object-oriented ones.

W14: Scientific Computing Environment: CORBA Components for the GRID

Organizers: Zsolt Kovács, László Varga

Day: Monday, 18 June

Today's scientific applications require access to clusters of computers, supercomputers, storage systems, data sources and different devices so researchers can work on a particular topic. The requirements to develop such applications are paramount. Not only are they distributed, heterogeneous and use different data sources, but they should be of high performance, maintainable, adaptable and user friendly. The programmer of such an application must be knowledgeable of many programming techniques (e.g. multi-threading, transaction, persistency) and has to know about distribution issues (e.g. global time, location, message ordering) as well as how to write algorithms. This can only be achieved by an environment which offesr extensibility and reuse through the synergetic integration of different tools (e.g. compilers, debuggers), libraries, executables.

Recently many scientific domains (e.g. High Energy Physics and Earth Observation) have decided to move towards a form of computing envisioned by the Cluster Computing (CC) community, called the GRID, as a possible solution. The GRID is analogous to the electric power grid and aims to couple geographically distributed data and computing resources and offer easy access to them.

On the other hand the Object-Oriented (OO) Computing community have come up with the idea of the Object Request Broker as an abstraction layer for distributed client-server applications. This led to the development of CORBA and Components (widely used in the industry) as the solution for extensibility and reuse.

Until now the CC and OO communities have developed their solutions independently but the organisers believe that both communities would benefit from a common discussion.

W15: Development of Robust and High Confidence Agent Applications

Organizers: Ciarán Bryce, Alexander Romanovsky, Anand Tripathi

Day: Monday, 18 June

The ECOOP Workshop on Mobile Object Systems is now in its 7th Year. The aim of the workshop series has been to bring together mobile object system and language designers, and generally people interested in discussing the current state and future direction of research in the mobile object system context. The workshop is a forum to learn about the latest research, and also to discuss and exchange ideas concerning ongoing theoretical and implementation work.

This year's workshop has two emphases. Firstly, it seeks experience reports, as well as papers on design and development techniques for mobile object applications. Application of the recent research results in the development of real systems is crucial for the future of mobile computing. Secondly, it brings together a group of active researchers working on security and fault tolerance to develop an understanding of the important research problems and recent results in these areas. In particular, it is felt that it should be beneficial to examine fault tolerance and security issues together as secure agents systems can be used for building fault tolerant systems and at the same time general fault tolerance mechanisms can be applied for providing security.

Our primary aim is to foster discussion between researchers and practitioners working in these areas. We are particularly interested in papers that challenge existing ideas and techniques in the mobile object field. Topics include, but are not restricted to:

Integration of agent and other technologies,
Mobility related experience reports and horror stories,
Mobile object/agent applications for the Internet,
Mobile object/agent application design and development techniques.
Interrelationships between security and fault-tolerance mechanisms in mobile object/agent systems,
Structuring techniques that facilitate provision of both fault tolerance and security,
Mechanisms for agent state and host resource protection,
Mechanisms for exception handling, error detection and recovery, checkpointing and roll-back.
Security and fault tolerance models for agent applications.

W16: Mining Pedagogical Patterns

Organizers: Joseph (Joe) Bergin, Jutta Eckstein

Day: Tuesday, 19 June

This workshop will bring together educators and industrial trainers interested in advancing the work of the Pedagogical Patterns Project. Participants will contribute patterns and patlets (pattern ideas) toward a refinement of the Experiential Learning Pattern Language (http://www.pedagogicalpatterns.org). The work of the group will be to refine and advance this language as well as to mine additional related patterns from the experience of the participants themselves.

The Experiential Learning Pattern Language has grown out of the work of many contributors to the Pedagogical Patterns Project, though it needs refinement and extension. The organizers have been key participants in this effort.

W17: The Next 700 Distributed Object Systems

Organizers: Eric Jul, Andrew Black, Rachid Guerraoui, Anne-Marie Kermarrec, Jochen Liedtke, Doug Lea, Salah Sadou

Day: Tuesday, 19 June

Over the last 15 years, the basic building blocks for distributed object systems have emerged: distributed objects, communicating with Remote Message Send (RMS), also known as Remote Method Invocation (RMI) or Location-Independent Invocation (LII). However, it has also become clear that such abstractions are by themselves sufficient to expose the hard problems of distributed computing, but not to solve them.

At last year's ECOOP workshop on Distributed Objects Programming Paradigms, we identified some of these problems (Security, Application Services, Other Communication Protocols, Tolerance of Partial Failures, Run-time Evolution, Meta-Object protocols and Ordering of events) that this year we will focus on.

The goal is to define and refine abstractions that address some of these problems and other like them. What are the right APIs, development methods, reasoning systems, and tools for building the next generation of Distributed Object Systems?

W18: 5th ECOOP Workshop on Quantitative Approaches in Object–Oriented Software Engineering

Organizers: Fernando Brito e Abreu, Brian Henderson-Sellers, Mario Piattini, Geert Poels, Houari A. Sahraoui

Day: Monday, 18 June

Measures of software internal attributes have been extensively used to help software managers, customers and users to characterize, assess, and improve the quality of software products. Many large software companies have intensively adopted software measures to increase their understandability of how (and how much) software internal attributes affect the overall software quality. Estimation models based on software measures have successfully been used to perform risk analysis and to assess software maintainability, reusability and reliability. However, most measurement efforts have focused on, what we call today, "legacy technology".

The OO paradigm provides more powerful design mechanisms. Much work is yet to be done to investigate analytically and/or empirically the relationships between OO design mechanisms, e.g., inheritance, polymorphism, encapsulation, usage, etc., and different aspects of software quality, e.g., modularity, modifiability, understandability, extensibility, reliability, reusability, etc. Furthermore, new technologies, e.g., UML, OO frameworks, OO Analysis/Design patterns, OO architectures, OO components, which take advantage of OO design mechanisms have been proposed in order to improve software engineering productivity and software quality. However, to better understand the pros and cons of these technologies on products developed using them we must be able to assess the quality of such products via adequate software product measures.

W19: Mechanisms for Enterprise Integration: From Objects to Ontologies

Organizers: Sergio de Cesare, Mark Lycett

Day: Tuesday, 19 June

The growth of the Internet and the widespread of e-commerce systems have accentuated pre-existing problems related with systems interoperability and the definition of common semantics throughout business organisations. Communication and exchange of data and information between business systems is nowadays dependent on the definition, modelling, design and implementation of underlying concepts shared across organisations. The issues involved are not just related to infrastructure and technology. Indeed fundamental problems exist around the understanding of what ?concepts? are shared, how they relate and what mechanisms should be adopted to allow systems to communicate and interoperate at all levels.

Genericity mechanisms, such as patterns, frameworks and components, have the potential for defining, modelling, designing and implementing shared concepts. Such mechanisms have inherited many object-oriented principles and built upon them to enhance the generalised nature of business problems and solutions both vertically (within the same domain) and horizontally (across different domains). Increasingly, these are supported by emerging ontological definitions and related languages (e.g. XML) designed to exemplify genericity.The potential contribution of XML to the definition of business ontologies and semantics is also relevant. Researchers and practitioners world-wide are exploring the possibility of defining interoperable industry specifications based on XML and the definition of an Ontology Markup Language.

The adoption and implementation of genericity mechanisms and ontological ?tools? is difficult to implement in greenfield situations. These difficulties are exacerbated in brownfield situations in which clear architectural separations do not exist and multiple technologies do exist.

In light of this the aim of the workshop is to discuss:

How shared business ontologies can be defined and represented;

How the underlying semantics can be represented and implemented through object and component-based technologies;

The role of mechanisms of generic abstraction in the definition of common semantics;

The role of XML in the definition of business ontologies.

The workshop will follow these phases:

Presentation of reviewed papers followed by discussion and questions;

Participants will be divided into groups of interest to discuss specific subtopics emerging from the previous phase;

General discussion of points drawn by the groups in phase two;

Conclusion.

ECOOP 2001

This workshop is addressed to all academics, researchers and practitioners interested in all approaches aimed at achieving the integration of enterprise systems. Papers submitted to the workshop will be reviewed by a committee of five members. The committee is composed of the workshop organisers, an external academic and two industrial practitioners.

W20: Engineering Complex OO Systems for e−Commerce

Organizers: Siobhán Clarke, Simon Dobson, Vinny Cahill

Day: Tuesday, 19 June

e-Commerce solutions tend to be complex applications displaying all of the intricacies of traditional systems, exacerbated by some additional issues. Advances are being made in software engineering support for complex development, such as component-based development, and advanced ways of separating concerns such as aspect-oriented programming, multi-dimensional separation of concerns, adaptive software, composition filters and much more. An interesting suite of object-oriented solutions has emerged, building on both mature infrastructures such as CORBA and on newer platforms such as Java and agent systems. Such techniques have been successfully been applied to developing the first generation of B2B and B2C e-commerce solutions. If more complex e-commerce solutions are to be engineered rather than simply built, however, we must confront a new set of issues. These include the creation and management of trust in open environments, the construction and destruction of distributed interactions over short timescales, the dependability of systems using untrusted code, the creation of standards in the face of competing business needs, and so forth. It is not clear how far standard object-oriented methods and techniques can successfully be extended to address these areas, and there is currently no forum available for their discussion.
Major topics:
Trust models and policies for open networksEngineering stable systems in the presence of untrusted code
Engineering for application scalability and 24x7 availability
Engineering models to support discovery, reflection and interoperability of components, and aggregation/integration of data from multiple (possibly unknown) sources
Separation of concerns within complex e-commerce systems
Frameworks and component engineering for e-commerce In addition, a new class of mobile applications bring new issues, including:
Engineering a tolerance of poor communications facilities
Engineering for exploitation of location-based information
Transcoding and repurposing content for multiple display types

W21: Workshop on Object–Oriented Business Solutions

Organizers: Rafael Corchuelo, Antonio Ruiz, Jörg Mühlbacher, Jesús D. García-Consuegra

Day: Monday, 18 June

E-commerce is gaining currency as the Internet settles as a medium for fruitful commercial transactions. Anyone with a credit card and an Internet-enabled device is a potential customer, and this is the reason why no corporation can resist jumping on the e-commerce bandwagon. To achieve this goal, web-based, attractive, technically robust applications are the key.

The e-world currently amounts to million Euros, and the underlying software industry is investing a lot of money in researching on new tools and development methods so that this kind of applications can be built at sensible costs. This has implied adapting existing methods and tools, which have usually been procedural, to the new resources and means the Internet provides, but at such a pace that they do not have enough time to consolidate before new proposals sprout out.

Object-orientation and component-based solutions seem to be promising and the trend towards incorporating them seems to be settling at an appropriate pace. At the same time, the complexity of these applications increases, and one of the reasons lies in the heterogeneous nature of the run-time and development-time aspects to be taken into account: architecture, security, attractiveness, quality of service, security, robustness, and so on.

The aim of the workshop is to discuss the research that is being carried out in universities and industries to solve the problems related to the construction of e-commerce applications using object-oriented technology, focusing on real-world industrial experiences and innovative infrastructure for building e-commerce solutions.

W22: 4th ECOOP Workshop on OO and Operating Systems

Organizers: Darío Álvarez Gutiérrez, Francisco Ballesteros, Paniti Netinant

Day: Tuesday, 19 June

The workshop aims to bring together researchers and developers working on object-oriented operating systems and to provide a platform for discussing problems arising from the application of object-orientation to operating systems and solutions for them.
Suggested topics for position papers and discussions include, but are not restricted to:
adaptable and adaptive OOOS,
frameworks for OOOS,
architecture of OOOS,
distributed OOOS and middleware,
aspect orientation and OOOS design,
what are the penalties of OO in OS and how to avoid them,
reflective OOOS,
OOOS tools,
reusability and interoperability of OOOS components,
OOOS configurability, maintenance, tuning and optimization,
OOOS for embedded systems,
real-time OOOS.
The programme of the workshop consists of a talk given by an invited speaker, position paper presentations, discussions, and a summary session. Presented papers will be published in a printed version (a technical report, probably) which will be distributed to all participants after the workshop.

Technical Programme

The main conference will take place in the Northern Block of Eötvös Loránd University from Wednesday 20 June to Friday 22 June.

It consists of 18 technical contributions, three invited talks and two panels.

Wednesday, June 20, 2001

8:00-9:00 Registration

9:00-9:15 *Opening Session*

9:15-10:15 *Keynote Address*
Charles Simonyi: Languages, Objects, and Intentionality

10:15-10:45 Break

10:45-12:15 *Session 1: Sharing and Encapsulation*

Session Chair: Jan Vitek

Capabilities for Sharing: A Generalization of Uniqueness and Read-Only

John Boyland, James Noble, William Retert

Sealing, Encapsulation and Mutability

Marina Biberstein, Yossi Gil, Sara Porat

Simple Ownership Types for Object Containment

David Clarke, James Noble, John Potter

12:15-13:45 Lunch

13:45-14:45 *Session 2: Type Inference and Static Analysis*

Session Chair: Satoshi Matsuoka

Distinctness and Sharing Domains for Static Analysis of Java Programs

Isabelle Pollet, Baudouin Le Charlier, Agostino Cortesi

Precise Constraint-Based Type Inference for Java

Tiejun Wang, Scot Smith

14:45-15:15 Break

15:15-16:15 *Session 3: Language Design 1*

Session Chair: Mira Mezini

CCC: User Defined Object Structures in C

Yasunori Harada Yamazaki, Richard Potter

Fickle: Dynamic Object Reclassification

Sophia Drossopoulou, Ferruccio Damiani, Mariangiola Dezani, Paola Giannini

17:00 *Welcome reception*

Thursday, June 21, 2001

9:00-10:00 *Invited Talk*

Erik Meijer: Language Interoperability in the .NET Technology

10:00-10:30 Break

10:30-12:00 *Session 4: Implementation Techniques*

Session Chair: Urs Hoelzle

A Quasi-Optimal Bit-vector Encoding of Tree Hierarchies. Application to Efficient Type Inclusion Tests

Olivier Raynaud, Eric Thierry

On the Usefulness of Liveness for Garbage Collection and Leak Detection

Martin Hirzel, Amer Diwan, Antony Hosking

Concurrent Cycle Collection in Reference Counted Systems

David Bacon, V.T. Raja

12:00-13:30 Lunch

13:30-15:00 *Session 5: Reflection and Concurrency*

Session Chair: Frank Buschmann

A Bytecode Translator for Distributed Execution of Legacy Java Software

Michiaki Tatsubori, Toshiyuki Sasaki, Shigeru Chiba, Kozo Itano

Reflections on MOPs, Components, and Java Security

Denis Caromel, Julien Vayssiere

The Optimistic Readers Transformation

Robert Strom, Joshua Auerbach

15:00-15:30 Break

15:30-17:00 *Panel A: The next 700 Distributed Object Systems*

Chair: Eric Jul

19:00 *Conference Banquet*

Friday, June 22, 2001

9:00-10:00 *Invited Talk*

Alistair Cockburn: People and the Limits of Methodology

10:00-10:30 Break

10:30-12:00 *Session 6: Language Design 2*

Session Chair: Markku Sakkinen

Family Polymorphism

Erik Ernst

An Overview Of AspectJ

Gregor Kiczales, Erik Hilsdale, Jim Hugunin,
Mik Kersten, Jeffrey Palm,William Griswold

True Modules for Java Classes

Davide Ancona, Elena Zucca

12:00-13:30 Lunch

13:30-14:30 *Session 7: Testing and Design*

Session Chair: Ana Moreira

Selecting an Efficient OO Integration Testing Strategy: An Experimental Comparison of Actual Strategies

Kamel Kamel, Vu Le Hanh, Yves Le Traon, Jean-Marc Jézéquel

Quality and Understandability in Use Case Models

Bente Anda, Dag Sjøberg, Magne Jørgensen

14:30-15:00 Break

15:00-16:30 *Panel B: To be Announced*

16:30-16:45 *Closing Session*

16:45 *Farewell Drink*

Keynote Address: Languages, Objects, and Intentionality

Presenter: Charles Simonyi

Abstract: The multiplicity of natural languages and the multiplicity of computer languages invite a false parallel. In fact computer languages have settled into an evolutionary rut that was carved by the needs of the earliest computers, while natural language evolves to even greater efficiencies. The talk will discuss "intentionality" as an important aspect of natural language, and the uses of intentionality in computer languages and in object oriented programming.

Presenter's profile: Charles Simonyi is Distinguished Engineer at Microsoft working on future products. His contributions included starting and managing the teams that developed Microsoft Excel, Multiplan, Word, and other software applications. Later, he lead a research team that established Intentional Programming as a new field of inquiry.

Before coming to Microsoft, Simonyi worked at Xerox Palo Alto Research Center developing Bravo, the first WYSIWYG (what you see is what you get) editor.

Simonyi, born in Budapest, Hungary, holds a bachelor of science degree from the University of California at Berkeley and a doctorate in computer science from Stanford. He has been elected to the National Academy of Engineering for "his contributions to the development of widely used desktop productivity software".

Simonyi has endowed chairs for "Public Understanding of Science" at Oxford University - presently held by Prof. Richard Dawkins; and for "Theoretical Physics" at the Institute for Advanced Study at Princeton - held by Prof. Ed Witten.

Invited Talk: Language Interoperability in the .NET Technology

Presenter: Erik Meijer

Abstract: Microsoft .NET extends the ideas of both the Internet and the operating system by making the Internet itself the basis of a new operating system. Ultimately, this will allow developers to create programs that transcend device boundaries and fully harness the connectivity of the Internet in their applications.

The "Common Language Runtime" (CLR) is key to the .NET framework. The CLR allows you to define a class in Visual Basic and inherit from it in Cobol, or to raise an exception in JScript and handle it in Perl. Compilers that support the CLR compile programs into the .NET intermediate language (MSIL). This intermediate code is then verified and JIT-ed by the runtime into executable code. The CLR also supports automatic garbage collection, cross-language debugging, and interoperability with classic COM and the Win32 platform. In this talk course we will give a birds-eye overview of the whole .NET framework and zoom in on several of its key features such as the architecture of the execution engine, the role of meta-data and self describing components in the runtime, and the in particular common language subset (CLS) of the runtime's type system.

Presenter's profile: Erik Meijer is a Program Manager in Common Language Runtime group at Microsoft and adjunct professor of Computer Science and Engineering at the Oregon Graduate Institute. Prior to joining Microsoft, while at Utrecht University, he has worked closely with researchers and developers of Microsoft and the Oregon Graduate Institute to integrate the lazy functional languages Haskell and Mondrian with the .NET framework.

Invited Talk: People and the Limits of Methodology

Presenter: Alistair Cockburn

Abstract: People have a nasty habit of ruining neatly drawn up methodologies. It's not that they intend to, it is just that people are packaged as "individuals" while methodologies are packaged in "roles". We shall explore principles that follow methodologies versus those that follow people, see how far methodologies can take us versus where we have to give in to the characteristics of people, and how to concoct an agile-but-sufficient methodology on the fly, using a view of software development as a cooperative game of invention and communication.

Presenter's profile: Alistair Cockburn is a Consulting Fellow at Humans and Technology. He has designed custom high-speed computer graphics hardware, led research in telecommunications software development techniques and consulted on object-oriented projects over the last 26 years. His

two books, "Surviving OO Projects" and "Writing Effective Use Cases" are widely read, and he is working on "Software Development as a Cooperative Game." In 1993 he designed for the IBM's Consulting Group the first OO methodology to be centered on use cases, responsibilities, design patterns and an ultralight process. In 1997 he was special advisor to the Central Bank of Norway, where he first designed the Crystal family of ultralight methodologies. In 2001 he helped coin the term "agile methodologies" to replace "light methodologies". Although he cares about methodology, he cares more about getting running software out the door, which leads him to the title to this talk. Alistair sometime remembers the six languages he is supposed to be able to speak, enjoys traveling, dancing, and sitting underwater.

Exhibits, Demonstrations and Posters

The main conference is accompanied by a three-day commercial Exhibitors' Forum from June 19 to 21. This exhibition is set up next to the main conference area and participants will thus have easy access to visit the stands even during the breaks. Participating companies and vendors will show and demonstrate their object-oriented products and services. Publishers and booksellers will display the most recent titles available in the field of object-orientation. If you want to be an exhibitor please contact the *Exhibit Chair* at `exhibit@2001.ecoop.org`.

Live demonstrations of the very latest object-oriented technology are an exciting part of every ECOOP conference, offering an excellent occasion for discussing technical aspects of object-oriented applications, tools and systems. These demonstrations are given by technical members of their implementation team on Wednesday, Thursday and Friday, in parallel with the technical program. To propose a demonstration please contact the *Demonstration Chair* at `demo@2001.ecoop.org`.

Poster sessions are an alternative forum for viewing results of object-oriented work, with the opportunity for one-to-one interaction with presenters. Poster themes cover the breadth of object-oriented technology - from theory to experiences in applications. They provide an easy and informal means for presenting ongoing research to the ECOOP 2001 audience. This may be especially useful for new ideas that have not yet been developed to the point of a regular paper. Posters will be on display during the entire conference. For information on submitting a poster please contact the *Poster Chair* at `poster@2001.ecoop.org`.

Useful to Know

Conference Venue

The conference will be held at Eötvös Loránd University, Budapest, Hungary. The university campus is located to the Southwest of the city centre, in district No. XI. See the campus map at the back of this booklet.

Travel info

If you arrive at Budapest by air, you land at Ferihegy Airport. We highly recommend you to use the Airport Minibus service for transfer to your hotel. If you arrive by train the best choice is the underground.

Internet Access

You can check your e-mail and contact your home base using the facilities in the computer rooms on the second floor of the Building of Informatics (see plan at the end of this booklet).

Meals and refreshments

In your registration package you receive luncheon tickets. Please do not forget to keep them with you. Coffee, tea and some refreshments will be served during the breaks. Refreshments and lunches are served only to delegates wearing badges.

Currency

The Hungarian currency is the Hungarian Forint (HUF). At the time of printing this document 1 euro is about 266 HUF.

Credit cards are accepted in most places in Budapest, and there are a lot of automatic cash dispensers and currency change offices in the city.

Electricity

Hungary uses 220V 50Hz electric power supply and you may need a special plug adapter to operate your electric appliances

Insurance

The conference organizers will not be able to take out any kind of insurance for the participants or partners. Participants are requested to make their own arrangements concerning insurance.

Visa

A valid identity card or passport is required for entry. Please contact the Hungarian Embassy or Consulate in your country for details on entry requirements (e.g. you may need a visa).

Hotline

In case you need some help after your arrival at Budapest call 06-30-854-0347 where an English speaking representative of us will answer and help.

Social Programme

Social Events

During the conference the following social events will be organized:

- Get-Together Party (Monday 18)
- Welcome Reception (Wednesday, 20)
- Conference Banquet (Thursday, 21)
- Farewell Drink (Friday, 22)

All social events are included in the conference fee, except for the Conference Banquet, which is included only for general registration. Participants of other registration types can attend the banquet for additional fee only.

Pre– and Postconference programmes

Our partner Pannonia Tourist Service offers several tours before and after the conference:

- Leisure at the Lake Balaton
- Exploring North East Hungary
- Sightseeing tour in Budapest
- Budapest by night
- Excursion to the Hungarian 'Puszta'

All programs include the costs of transportation, English speaking guide, entrance fees where applicable. Programs will be arranged in case of minimum 12 participants.

In addition Pannonia offers extra activities for accompanying persons. Details about them are on our website.

Accommodation

Hotel allocation is strictly on a first-come, first-served basis. In the list of hotels below all prices are shown in Euro. Detailed information about hotels (location, price, facilities, etc.) can be found on our website.

Hotel	*Single room*	*Double room*
Hotel Hyatt Regency*****	225	240
Hotel Art Plaza****	135	158
Hotel Gellért****	89	194
Hotel Mercure Korona****	117	133
Hotel Páva****	77	82
Hotel Mercure Metropol***	77	87
Hotel Mercure Relais Duna***	69	75
Hotel Ibis Centrum***	69	75
Hotel Griff**	43	49
Hotel griff Junior*	24	33
Summer Hotel Hill	-	20/person

Hotel reservation is done through the on-line registration form of the ECOOP 2001 website.

Registration and Conference Fee

To register visit our website at www.2001.ecoop.org. The registration consists of the following steps:

- Fill the on-line form and submit it.
- Check the resulting registration summary. From here you can go back to the online form to make modifications if needed. If your summary is OK, submit it.
- Print the resulting payment form, fill the required information (e.g. credit card no.) and fax it to the number shown on the form.

Registration fees are differentiated by the time and type of registration. The time categories are: **early** (before May 15, 2001), **late** (after May 15, 2001) and **on-site**. The type of registration can be **general** (includes all services during the conference including banquet), **regular** (includes all services during the conference excluding banquet), **reduced** (for Eastern European participants, excluding

banquet), and **student*** (for students, excluding banquet). The following participation categories exist: **whole conference** (includes access to the workshops, lunches and coffee breaks during the whole conference, the conference itself and all evening events), **workshop only** (access to the workshop sessions, the corresponding day(s) workshop meals and coffee breaks and the Get-together party on Monday), and **accompanying person** (includes participation in the following social events: Get-together party, Welcome reception, and Farewell drink).

*Student type registration requires the confirmation of student status.

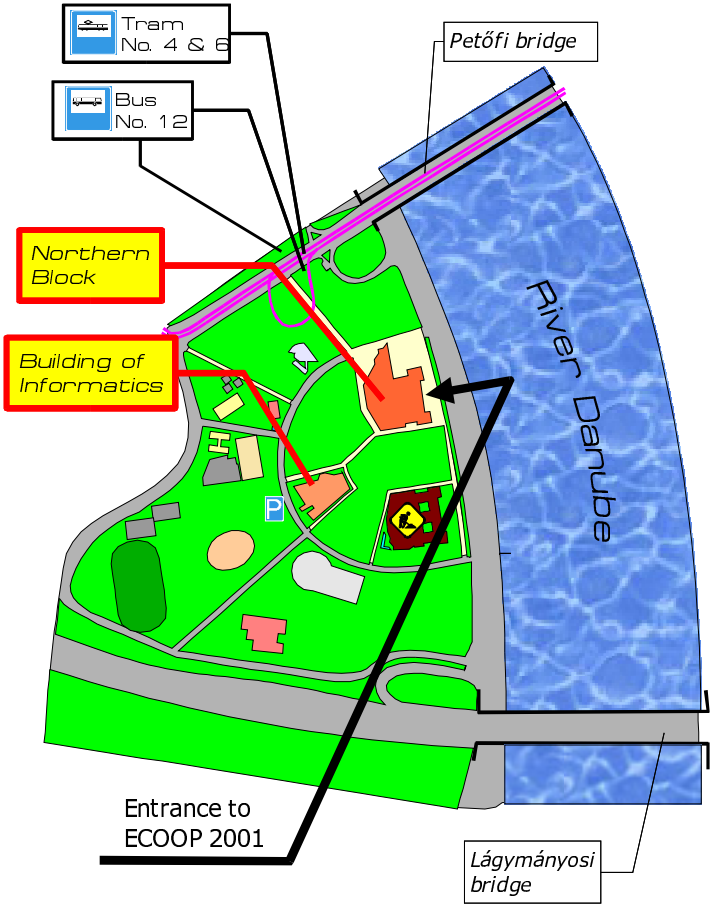
Registration fees in Euro (€)

	Whole Conference				*Workshop Only*			*Accompanying*
	general	*regular*	*reduced*	*student*	*regular*	*reduced*	*student*	
early	470	440	280	160	190	120	70	60
late	550	520	360	190	230	140	80	70
on-site	590	560	390	190	260	160	100	80

Additional tutorial fees in Euro (€)

	Number of units	*general or regular*	*reduced*	*student*
early	1 unit	190	110	80
	2 units	325	190	130
	3 units	455	260	170
	4 units	530	320	210
late	1 unit	245	140	90
	2 units	415	230	160
	3 units	590	310	210
	4 units	685	380	260
on-site	1 unit	285	170	120
	2 units	450	260	190
	3 units	630	340	240
	4 units	720	410	290

Campus Map



The campus address is: Budapest XI. Pázmány Péter sétány 1.

Organizers

Association Internationale pour les Technologies Objets

University Eötvös Loránd, Budapest

Sponsors

Contact

Tutorial Chairs	tutorial@2001.ecoop.org
Workshop Chairs	workshop@2001.ecoop.org
Panel Chair	panel@2001.ecoop.org
Poster Chairs	demo@2001.ecoop.org
Demonstration Chairs	poster@2001.ecoop.org
Exhibit Chairs	exhibit@2001.ecoop.org

Official website

www.2001.ecoop.org