9th International Conference on
RELIABLE SOFTWARE TECHNOLOGIES
ADA-EUROPE 2004
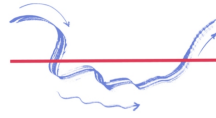

PALMA DE MALLORCA, SPAIN, JUNE 14-18, 2004

ADVANCE PROGRAM

http://www.ada-europe.org/conference2004.html

Presentation

In 2004, the 9th International Conference on Reliable Software Technologies takes place in Palma de Mallorca, Spain, from June 14th to June 18th. The conference offers a technical program and an exhibition, plus a series of tutorials.

The conference offers an international forum for researchers, developers and users of reliable software technologies. Presentations and discussions cover applied and theoretical work currently conducted to support the development and maintenance of software systems.

Palma de Mallorca is a city that has had a university since the 14th century, although in 1715 the new Borbonic king suppressed the universities in both Barc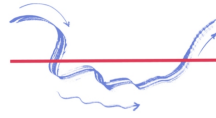elona and Palma as a punishment for their opposition to the dynasty. Barcelona had to await until 1842 to see its university restored. Mallorca had to await even further: until 1978.

The port of Palma had played an important role in the commerce of the early Catalan empire from the 13th to the 15th centuries and some well preserved Gothic traces still witness this splendorous past, remarkably the Cathedral and the Almudaina palace.

The beauty of the countryside and its privileged climate has converted the island in an attractive destination for leisure since the 19th century. Such a beauty still remains, although in a fragile state, due to the destructive effects of the mass-oriented tourism industry.

Overview of the Week

	Morning	Late Morning	After Lunch	Afternoon
Monday **June 14th** Tutorials	J.-P. Rosen *Developing a Web Server in Ada with AWS*			
	M. Heaney *Programming with the Charles Container Library*		G. Bernat *Probabilistic Worst Case Execution Time Analysis*	
	M. A. Alves *No Pointers, Great Programs*		A. Strohmeier *Requirements Analysis with Use Cases*	
Tuesday **June 15th** Sessions & Exhibition	S. Tucker Taft *Fixing Software before It Breaks: Using Static Analysis to Help Solve the Software Quality Quagmire*	Static Analysis	Distributed Systems	Vendor sessions
				Ada-Europe General Assembly
Wednesday **June 16th** Sessions & Exhibition	M. Gogolla *Benefits and Problems of Formal Methods*	Real-Time Systems	Testing	P. Leroy *An Invitation to Ada 2005*
		Reflection and XML		
Thursday **June 17th** Sessions & Exhibition	A. Olivé *On the Role of Conceptual Schemas in Information Systems Development*	Critical Systems Modelling	Scheduling	S. Vinoski *Can Middleware Be Reliable?*
			Application Programming Interfaces	Closing Session and Awards
Friday **June 18th** Tutorials	P. Amey and R. Chapman *Practical Experiences of Safety and Security-Critical Technologies*			
	B. Lewis and E. Colbert *Developing Fault-Tolerant, Time-Critical Systems with AADL, UML and Ada*			
	B. Brosgol *Real-Time Java for Ada Programmers*			

INVITED SPEAKERS

Fixing Software Before It Breaks: Using Static Analysis to Help Solve the Software Quality Quagmire

S. Tucker Taft
President, SofCheck, Inc.

Tuesday June 15th, 9:00

There is growing realization that something must be done to improve the quality and security of software, but the costs associated with exhaustive testing approaches are economically prohibitive for all but the most critical systems.

This talk will discuss the progress being made to develop scalable and accurate static analysis technologies to help find a way out of the current software quality quagmire. Static analysis has the potential to automate the process of line-by-line, full path inspection of source code for defects, while also providing component-by-component characterization of the software system in terms of inputs, outputs, and effects, thereby fostering better understanding of legacy software and enabling more informed software evolution and reuse strategies.

In the real-time arena, static analysis has been used to detect race conditions and possible deadlocks, and help with schedulability determination. When coupled with this more general capability for automated defect detection and component characterization, new opportunities are created. In particular, by implicitly adding to the source code metering variables to count quantities such as stack depth, loop counts, dynamic storage allocation, lock frequency and duration, etc., the same static analysis technology can summarize these additional kinds of important real-time characteristics of the code, and help in the process of turning real-time system development from a black art into a true engineering discipline.

Short Biography

S. Tucker Taft is President and Founder of SofCheck, Inc., a company devoted to providing tools and technologies to help improve the quality of software through automated bug detection and web-based development team collaboration. Mr. Taft graduated Summa Cum Laude from Harvard College in 1975, and then worked four years running the Harvard student computer center, one year as a consultant, and twenty-two years for Intermetrics, Inc. (and its successors AverStar and AverCom).

From 1990 to 1995, Mr. Taft led the Ada 9X language design team, culminating in the February 1995 approval of Ada 95 as the first ISO standardized object-oriented programming language.

Benefits and Problems of Formal Methods

Prof. Dr. Martin Gogolla
Department for Computer Science, University of Bremen, Germany

Wednesday June 16th, 9:00

Formal methods for software development have been discussed for decades. The enormous spectrum ranges from naive mathematical set theory over various approaches for model or property oriented specification to newer developments like the Object Constraint Language (OCL) being an integral part of the Unified Modelling Language (UML). People's opinions on formal methods are naturally determined by their experience and knowledge on the subject. This paper will explain when and under what circumstances formal methods and languages like OCL can be employed in a beneficial way. It will also try to point out when a formal approach should not be taken. The success of using a formal approach is highly influenced by the expectations and pre-requisite knowledge of the developers, the role the formal support in the development process is given, and of course by the used tools.
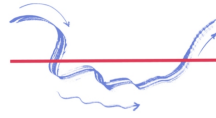
Short Biography

Martin Gogolla is professor for Computer Science at University of Bremen, Germany and is the head of the Research Group Database Systems.

His research interests include object-oriented design, formal methods in system design, semantics of languages, and formal specification. Before joining University of Bremen he worked for the University of Dortmund and the Technical University of Braunschweig. His professional activities include: Teaching computer science; publications in journals and conference proceedings; publication of two books; speaker to university and industrial colloquia; referee for journals and conferences; organizer of workshops and conferences (e.g. the UML conference); member in national and international program committees; contributor to international computer science standards (OCL 2.0 as part of UML 2.0).

On the Role of Conceptual Schemas in Information Systems Development

Antoni Olivé
Politechnical University of Catalonia, Spain

Thursday June 17th, 9:00

In the literature, the role of Conceptual Schemas (CS) and conceptual modelling in Information Systems (IS) development is not well established. On the other hand, there is some confusion over the relationships between CSs and similar concepts such as domain knowledge, functional specifications and ontologies. This talk will try to shed light on these issues. We show that a CS is the knowledge needed by an IS to perform its functions. We then analyse the possible roles of CSs in the IS architecture, which range from specification to explicit software component. We also show the role of CSs in the context of the recent OMGs MDA. The paper focuses on ISs, but most of the conclusions may apply to the general field of software.

Short Biography

Antoni Olivé is a professor of information systems at the Universitat Politècnica de Catalunya in Barcelona. He has worked in this field during over 20 years, mainly in the university and research environments. His main interests have been, and are, conceptual modeling, requirements engineering, information systems design and databases. He has taught extensively on these topics. He has also conducted research on these topics, which has been published in international journals and conferences. He was the recipient of the DKE-ER award for his contribution to the *Intl. Conf. on Conceptual Modeling (ER '2003)*. He is a member of IFIP WG8.1 (Design and evaluation of information systems) where he served as chairman during 1989-1994.

Can Middleware Be Reliable?

Stephen Vinoski
Chief Engineer of Product Innovation for IONA Technologies in Waltham, MA.

Thursday June 17th, 16:00

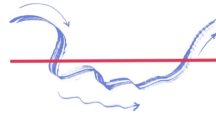Middleware is everywhere, providing adaptability between systems and layers, providing the needed connectivity within and between enterprise computing systems, gluing everything together. In this talk, Steve will first present a brief history of middleware and describe the origins of different types of middleware. He'll then discuss the current state of Web Services middleware, with an emphasis on standardization efforts related to reliability.

Short Biography

Steve Vinoski is Chief Engineer of Product Innovation for IONA Technologies in Waltham, MA. Steve is also an IONA Fellow. As of the start of 2003, Steve has authored or co-authored over 40 highly-regarded publications about distributed computing, including the book "Advanced CORBA Programming with C++," widely acknowledged as the "CORBA Bible."

He has written the popular "Object Interconnections" column on distributed object computing for the C/C++ Users Journal (and formerly for SIGS C++ Report) since 1995 with Dr. Douglas C. Schmidt. He also writes the "Toward Integration" middleware column for the IEEE Internet Computing magazine.

Tutorials

Developing a Web Server in Ada with AWS

J.-P. Rosen, Adalog, France

Monday June 14th, full day

This tutorial describes AWS, the Ada Web Server, and how to use it for the development of web applications. It describes the principles of AWS, from the most basic functionnalities to the more advanced ones (Authentication, SOAP interface, session management, hotplugs, multi-server applications, etc.) The seminar emphasises practical usage of AWS, and presents design patterns that have proved effective for developping existing applications. It compares the development process with AWS to other techniques.

The tutorial provides attendees with the information needed to assess whether AWS is appropriate to their needs, and the necessary knowledge to start writing full-scale Web applications.

Presenter

J.-P. Rosen is a professionnal teacher, teaching Ada since 1979 (no typo here, it was preliminary Ada!). He runs Adalog, a company specialized in providing training, consultancy, and services in all areas connected to the Ada language. He has developed a complete application with AWS for Adalog's internal usage (paper presented at SIGAda 2003).

Why should you participate in this tutorial?

Because AWS represents a new advance in Web applications development. Rather than having a heavy-wieght server like Apache, and scripts in various languages, it allows the development of stand-alone web servers dedicated to a single purpose, fully written in one language: Ada.

Although AWS can be used to develop conventional Web applications, it opens up new possibilities: Regular applications that use a Web interface and a browser as the GUI; Distributed applications that communicate through an HTTP link; Client/server applications that need to be accessible by several users at the same time; Mixed applications, where the web interface is only part of the job, like a control/command system which is remotely monitored through a web interface, and more!

Participants to this tutorial will not only learn how to use AWS, they will discover a new approach and new possible solutions that are applicable to a wide range of applications.

Programming with the Charles Container Library

Matthew Heaney, On2 Technologies, US

Monday June 14th, morning

Charles is a container library for Ada95 that elegantly solves many typical programming problems. The library comprises sequence containers (vectors, deques, and linked-lists) for inserting elements at specified positions, and associative containers (sets, multisets, maps, and multimaps) which position elements in order by key. Charles is general, flexible, and efficient, and its design has been guided by the philosophy that a library should stay out of the programmer's way.
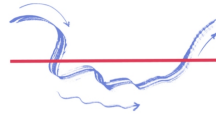
Presenter
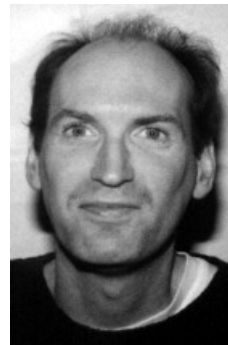
M. Heaney is the author of the Charles library. His interests include object-oriented programming, design patterns, and library design. He has recently submitted a proposal based on Charles to the ARG in response to AI-302 calling for a standard container library API.

Why should you participate in this tutorial?

The C++ STL is an innovative container library that has become quite popular. The Charles container library is essentially a port of the STL to Ada95, and by attending this tutorial you'll learn about the STL and understand why it is important. One of its primary design features is that it uses iterators to abstract-away container differences, allowing you to use purely generic algorithms that operate over any container. The time semantics of each container operations are specified rather than being an implementation detail, so that you can make guarantees about run-time behavior. The sequence containers are general enough to obviate the need for traditional data structures as stacks or queues, and the associative containers perform fast dictionary-style lookups. A proposal based on Charles has also been submitted for standardization in the next revision of the Ada language.

Probabilistic Worst Case Execution Time Analysis

Dr. G. Bernat, Department of Computer Science, University of York, UK

Monday June 14th, morning

A large number of problems in the development and deployment of embedded real-time systems have its roots on timing problems. The approach of end-to-end measurement of the code is error prone and costly, but most importantly, it may be unsafe as the method may fail to accurately find potential long execution times that may result in timing errors found too late in the development cycle or even after deployment. The tutorial will present the theory and tools for the determination of the worst-case execution time of programs based on an integration of measurement and static timing analysis.

Traditional static analysis techniques try to find the longest path in a program by analysing the code only (without running it). They may find a safe upper bound of the value, however for modern processors the level of pessimism is such that may result in useless results. On the other hand, end to end measurement observes the execution time by actually running the programs under different test conditions. The measurement based WCET analysis combines the best features of both approaches by determining the execution time of small program sections by direct observation through measurement, but using static analysis techniques to determine analytically the longest path of a program. In addition, the approach allows not only to provide a single value of the WCET but a probability distribution of the possible WCET values.

The tutorial will illustrate the main concepts of probabilistic WCET analysis and demonstrate it through a case study using the MarteOS Real-Time operating system using the pWCET toolset. The tutorial provides attendees with the information needed to assess whether AWS is appropriate to their needs, and the necessary knowledge to start writing full-scale Web applications.

Presenter

Dr. Guillem Bernat graduated in 1992 in Computer Science from the Universitat de les Illes Balears. He later

got his PhD from the same University in 1998. He was a lecturer at its Department of Computer Science from 1992 until 1999 when he was appointed as a research fellow at the Department of Computer Science at the University of York. In 2000 he was appointed as Senior Research Fellow. Since 2001 he is a lecturer at the same department. He is also Managing Director of Rapita Systems Ltd., a spin-off company of the University of York.

Guillem Bernat is interested in wide variety of aspects of real-time systems, in particular all issues related to probabilistic analysis of real-time systems including timing analysis, scheduling algorithms and their analysis and flexible scheduling architectures.

No Pointers, Great Programs. How to Stay on the Value Semantics Side of the Ada Way with the Help of Containers

## Mário A. Alves, University of Porto, Portugal

Monday June 14th, afternoon

How would you like writing complex object-oriented programs without the words **new**, **access**, **null**, or **free**? Well, there is a way —the Ada way—, and that's what we'll learn in this tutorial.

Pointers are evil. Although Ada has succeeded in averting many of their dangers (in relation to other languages), pointers in Ada (access types and values) are still considered a nuisance to say the least in many situations. For example, heterogenous arrays, ragged arrays, recursive types: you cannot write these things strictly in Ada without using pointers. But you can have them *encapsulated*, inside the proper container type-and then forget about them and get on with your life of writing great programs. We'll study several of these magical containers, starting with the wonderful Unbounded_String —which we'll take as our paradigmatic example—, and moving on to fully fledged container libraries like Charles. We'll also review the Ada foundations for value semantics (e.g. unconstrained types), strict Ada 95 containers (arrays, strings, files), generics, and class-wide programming.

Presenter

Mário A. Alves is currently with the University of Porto, doing PhD research on adaptive hypertext. He
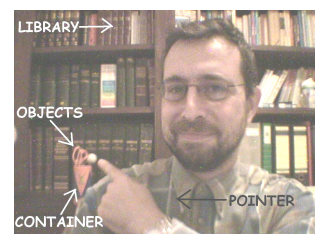
has degrees in Linguistics and Informatics Engineering. He has taught Ada, Software Engineering and other courses at the New University of Lisbon, Open University of Portugal, and other places. He has been a software developer since the mid 1980's and an Adaist since the mid 1990's. He has been on the Ada Standard Container Library Working Group since its inception in the Ada-Europe 2002 workshop. Lately he has been a contributor to Ada Issue 302 for the creation of such a library for Ada 2005, where he has focused on indefinite elements and persistence.

Why should you participate in this tutorial?

Because pointers are evil, and although they are safer in Ada than in other languages, they are still a nuisance in many situations (see abstract). Fortunately, we can escape the pointer tyranny with the help of properly designed container libraires and a focus on value semantics —an area where Ada also excels. If you're a newcomer to the Ada way, you'll benefit from starting your journey on the value semantics side of it. If you're an experienced Adaist, unsatisfied with the pointer idiom, you'll want to try a new approach. In any case: because you want to write great programs without the words **new**, **access**, **null**, or **free** —and therefore with increased understandability and safety.
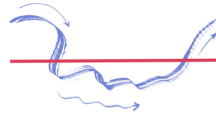
Requirements Analysis with Use Cases

A. Strohmeier, Software Engineering Laboratory, Swiss Federal Institute of Technology in Lausanne (EPFL), Switzerland

Monday June 14th, afternoon

We will show how Use Cases can be used at an early stage of the software development process for modeling user requirements and getting insights into the expected behavior of the system.

A use case describes every possible situation that can arise when a user has a particular goal against the system. Use cases are an excellent tool for capturing behavioral requirements of software systems. Use cases are popular because of their informal, easy to use and to understand style which caters to technical as well as non-technical stakeholders.

Contrary to popular belief, use cases are primarily textual descriptions, whereas he graphical appearance, called a use case diagram in UML, tells nothing more than the names of the use cases and their relationships to actors.

Upon completion of this tutorial, participants should be able to:

- understand the role of use cases in requirements analysis;

- understand the importance of capturing the functional requirements without going into design/implementation detail;

- understand the relationship between use cases and non-functional requirements;

- understand what makes an effective use case;

- understand the limitations of use cases and be aware of other models available that can make use cases more precise and rigorous.

Presenter

Alfred Strohmeier is a Professor of Computer Science at the Swiss Federal Institute of Technology in Lausanne (EPFL), Switzerland, where he leads the Software Engineering Laboratory. He has been teaching object-oriented technologies and Ada in academic and industrial settings. He was a Distinguished Reviewer of Ada 95 and participated in its definition. In 2003, he became a member of the Swiss Academy of Engineering Sciences. His current interests are software engineering; software development methodologies, especially object-oriented approaches; software development environments; and technologies related to the Ada language.

Why should you participate in this tutorial?

You should attend this tutorial if you want to learn how to write effective Use Case descriptions for requirements elicitation. It is often difficult to bootstrap the software development process in an orderly manner. The first step, i.e. requirements elicitation, is mostly about communication between people. When some of the stakeholders are not software specialists, technical notations are clearly not suitable as a communication vehicle. Because of their informal nature, Use Cases are then a popular choice. A complete set of Use Cases specifies all the different ways to use the system, and thus defines all behavior required of the system, bounding the scope of the system. Even though Use Cases are a powerful tool, some training and discipline are needed for writing effective Use Cases. The tutorial proposes to use a goal-oriented approach, and will propose a format for writing Use Case descriptions and criteria for structuring them. The approach will be exemplified on several case studies.

Practical Experiences of Safety and Security-Critical Technologies

P. Amey & R. Chapman, Praxis Critical Systems, UK

Friday 18th, full day

The tutorial identifies the special properties of systems intended for use in ultra-reliable domains and the qualitative shift in development methods that is required to achieve those properties. The advantages (and weaknesses) of Ada are introduced in the contexts of the ISO HRG report on High-Integrity Ada and of the SPARK sub-language. The demands of common, important development standards are described together with appropriate and cost-effective techniques for meeting them. Finally project experience illustrating successes in meeting the main standards is discussed.

Presenters

Peter Amey is an aeronautical engineer by original professional training and achieved Chartered Engineer status through the Royal Aeronautical Society. He served as an engineering officer in the Royal Air Force and spent several years at the Boscombe Down test establishment working on the certification of aircraft armament systems. Peter joined Program Validation Limited to develop the high-integrity language SPARK and its support tool the SPARK Examiner and continues that work today with Praxis Critical Systems. As well as developing SPARK he has used it on major programmes including Tornado, Eurofighter and the Lockheed C130J.
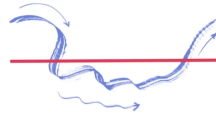
Peter teaches SPARK and Ada on a regular basis and has lectured widely on the development of critical systems. Most recently this has included the keynote address "Logic versus Magic" at Ada Europe 2001, "Closing the Loop: the Influence of Code Analysis on Design" at Ada Europe 2002, "A Language for Systems not Just Software" at SIGAda 2001 and "High Integrity Ravenscar" at Ada Europe 2003. Peter has also had a well-received article published in Crosstalk Journal.

Roderick Chapman received MEng and DPhil degrees from the University of York, England in 1991 and 1995 respectively. He is currently products manager at Praxis Critical Systems, leading the design and development of the SPARK language and toolset. Before joining SPARK team, Rod was involved in the implementation high-integrity real-time and embedded systems, including SHOLIS (the first system implemented to the Def Stan 00-55 SIL4 standard), the Lockheed Martin C130J Mission Computer, and the MULTOS CA. Rod has presented tutorial, papers and panel sessions at many conferences, including SIGAda, Ada Europe, and STC, and remains a member of the Ada95 HRG.

Why should you participate in this tutorial?
Because the presenters, and their company, Praxis Critical Systems, have an exceptional level of experience in the development of safety- and security-critical systems. Their experience spans aviation and rail in the safety domain as well as financial systems in the security domain; they have developed systems to meet all of the principal standards such DO-178B, Def Stan 00-55 and Common Criteria. The tutorial provides a unique opportunity to compare development approaches, their relationships with the various standards and to discover which approaches prove most cost-effective in practice

Developing Fault-Tolerant, Time-Critical Systems with AADL, UML, and Ada

B. Lewis, Research, Development and Engineering Laboratory, US Army's Aviation and Missile Command, USA

E. Colbert, Computer Science Department, University of Southern California, USA

Friday 18th, full day

The Society of Automotive Engineers (SAE) is developing a standard Architecture Analysis and Design Language (AADL) using UML and Honeywell's MetaH architecture language as a base. (Voting on the standard is expected to start in the first quarter of 2004.) MetaH has been used on 20 demonstration projects. The AADL specification defines an AADL profile for the Object Management Group's (OMG) Unified Modeling language (UML).

The AADL is designed to support the specification and analysis of performance-critical systems including real-time, fault-tolerant, safety-critical, securely partitioned, dynamically reconfigurable multi-processor systems, and system-of-systems architectures. The AADL enables the development of highly evolvable systems. It supports formal analyses of a system's architecture (integrated hardware and software components) early in the development. The AADL allows rapid, incremental development of prototypes that can evolve directly into the final production system.

In this tutorial, you will learn the concepts behind the AADL, the basic syntax and semantics of the textual and the UML-based graphical languages. The tutorial will also look at how to develop embedded, time-critical systems that can evolve as requirements or technology change, using AADL. The tutorial will show how a system can be designed using the AADL; analyzed for safety, timing, and reliability; and implemented using Ada.

Presenters

Bruce Lewis is a senior experimental developer for the US Army's Aviation & Missile Command, Research, Development and Engineering Laboratory, Software Engineering Directorate (SED). His work has focused on software architecture, reuse, and system evolution. His group has been experimenting with using Architecture Description Language for analysis and development since 1993. He has served as the government lead on various DARPA projects related to ADLs and real-time systems, including those developing the baseline for the AADL. He is the chairman of the AADL standardization subcommittee.

Ed Colbert is consulting with U.S. Army Aviation & Missile Command, and Honeywell Technology Center, on the definition of the Avionics Architecture Description Language (AADL) for the Society of Automotive Engineers (SAE). He is also lecturing in the Computer Science Department of the University of Southern California on software engineering and architectures, contributing to the Model-Based Software Engineering (MBASE) method of the USC Center for Software Engineering, and researching costing estimation for secure system development. Ed has been teaching object-oriented methods, software engineering, and the Ada programming language since 1982, and since 1986 consulting as well. He created the Colbert Object-Oriented Software Development method ("OOSD"), which supports analysis and design for implementation in languages such as Ada, C++, and SmallTalk. NASA
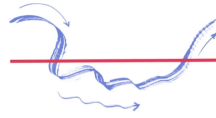
Langley Research Center used OOSD for a Software Engineering Process manual, chosen partly for its strength in real-time software development.

Why should you participate in this tutorial?

Because the AADL supports Model-Driven Architecture-Based Development which allows a system-engineering approach to designing the system software/hardware architecture, implementing it, and evolving it. This is a significant improvement over the empirical approaches currently used. There is strong interest in and participation in the language development in Europe and in the US. The European Space Agency has recently proposed a major research and tool development effort using the AADL. The COTRE research project on advanced computer system engineering for avionics systems, lead by Airbus, has adopted the AADL, as presented by Airbus at the last AdaEurope Conference in Toulouse.

Real-Time Java for Ada Programmers

Dr. B. Brosgol, Ada Core Technologies, USA

Friday June 18th, full day

Although the term "real-time Java" may sound self-contradictory, serious technical activity has been underway since early 1999 on extending the Java platform to satisfy the requirements for real-time systems, and several implementations exist. This work is relevant to the Ada community as both a challenge and an opportunity: on the one hand, it may compete with Ada in the real-time marketplace, but on the other hand some of its ideas may be worthy of consideration in a future version of the Ada language.

This tutorial will focus on the Real-Time Specification for Java ("RTSJ"), which was developed by the Real-Time for Java Expert Group under the auspices of Sun Microsystems' Java Community Process. The tutorial will analyze/critique the Java platform with respect to real-time support, summarize/illustrate the main elements of the RTSJ, and compare/contrast the design with Ada's real-time features (both in Ada 95 and under consideration for Ada 05). The tutorial will also outline the main aspects of the J-Consortium's "Core Extensions" (a competing real-time Java approach), will summarize a proposed high-integrity profile for the RTSJ, and will provide a status update on the real-time Java work and its usage and prospects.

Presenter

Dr. Ben Brosgol has over 25 years of experience in the computer software industry, with a focus on programming languages, software development methods, and real-time systems. He was a primary member of the Real-Time for Java Expert Group and a coauthor of the Real-Time Specification for Java. He is currently a member of the Technical Interpretations Committee for the RTSJ, and he has also served as a reviewer of the J-Consortium's Core Extensions. He has been delivering Java-related tutorials and courses since 1997.

Dr. Brosgol is an internationally-recognized expert on Ada. He participated in both the initial language design and the Ada 95 revision, and he is the past chairman of the ACM's Special Interest Group on Ada (SIGAda). He has published numerous papers on Ada, has delivered presentations and tutorials at many Ada Europe and SIGAda conferences, and has been conducting courses on real-time programming in Ada since the late 1980s. Most recently he was an invited keynote speaker at the 2003 SIGAda conference, where his topic was Ada and Real-Time Java: Cooperation, Competition, or Cohabitation? He is a senior member of AdaCore's technical staff in the US, in the Boston area.

Why should you participate in this tutorial?

Because you will learn the pros and cons of the Java thread model, both in general and for real-time applications. You will see how real-time Java addresses the apparent "show stopper" problem of garbage collection. You will be able to judge whether real-time requirements can be met by a "pure" Object-Oriented Language. You will understand the effect of a dynamic and flexible scheduling approach, in terms of expressibility, predictability, and performance. Finally you will also discover who is using real-time Java, and for what sorts of applications.

Conference Schedule
Preliminary version

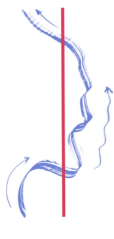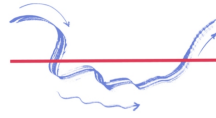|	Tuesday 15th	Wednesday 16th		Thursday 17th
9:00–10:00	Invited Talk: S. Tucker Taft, SofCheck, USA *Fixing Software Before It Breaks. Using Static Analysis to HelpSolve the Software Quality Quagmire*	Invited Talk: M. Gogolla, University of Bremen,Germany *Benefits and Problems of Formal Methods*		Invited Talk: A. Olivé, Politechnical University of Catalonia *On the Role of Conceptual Schemas in Information Systems Development*
10:00–11:00	Exhibition Opening & Coffee	Exhibition & Coffee		Exhibition & Coffee
	Static Analysis	Real-Time Systems	Reflection and XML	Critical Systems Modelling
11:00–11:30	P.T. Breuer, M. García-Valls *Static Deadlock Detection in the Linux Kernel*	J. Zamorano, A. Alonso et al. *Implementing Execution-Time Clocks for the Ada Ravenscar Profile*	P. Rogers, A. Wellings *Open Ada: A Reflective Compiler for Ada95*	P. Amey, N. White *High Integrity Ada in a UML and C World*
11:30–12:00	R.E. Sward *Extracting Ada95 Objects from Legacy Ada Programs*	M. Masmano, J. Real et al. *Extending the Capabilities of Real-Time Applications by Combining MaRTE OS and Linux Kernels*	Z. Vrandečić, D. Simon *XML4Ada95, Accessing XML using the DOM in Ada95*	H. Hagenauer, N. Martinek, W. Pohlman *Ada Meets Giotto*
12:00–12:30	J. Blieberger, B. Burgstaller, B. Scholz *On the Tree Width of Ada Programs*	A. Burns *Supporting Deadlines and EDF Scheduling in Ada0Y*		A.J. Hilton, J.G. Hall *High-Integrity Interfacing to Programmable Logic with Ada*
12:30–14:00	Lunch	Lunch		Lunch

CONFERENCE SCHEDULE (cont.)

Preliminary version

	Tuesday 15th	Wednesday 16th	Thursday 17th	
	Distributed Systems	**Testing**	**Scheduling**	**Application Programming Interfaces**
14:00–14:30	J. López-Campos, J.-J. Gutiérrez, M. González-Harbour *The Chance for Ada to Support Distribution and Real-Time in Embedded Systems*	S. K. Giri, Y. V. Jeppu, K. Karunakar *A Randomised Test Approach to Testing Safety Critical Ada Code*	J. Real, A. Burns et al. *Dynamic Ceiling Priorities: A Proposal for Ada0Y*	M.A. Alves *A Theory of Persistent Containers and its Application to Ada*
14:30–15:00	Th. Vergnaud, J. Hugues, L. Pautet, F. Kordon *PolyORB: A Schizophrenic Middleware to Build Versatile Reliable Distributed Applications*	K. P. Chan, T. Y. Chen, D. Towey *Good Random Testing*	A. Espinosa, A. García-Fornes et al. *Mixing Scheduling Policies in Ada*	M. Heaney *Shortcuts: A Critical Look*
15:00–15:30	S. Nardrchal *Event Language for Real-Time On-the-Fly Control According to the Initial Requirements*	B. Álvarez, J.-A. Pastor et al. *Teaching Real-Time Systems around a Digital Model Railroad Platform using Ada*	M. Aldea, J. Miranda *Implementing an Application-defined Scheduling Framework for Ada Tasking*	F. Gasperoni *Vector Processing in Ada*
15:30–16:00	Exhibition & Coffee	Exhibition & Coffee	Exhibition & Coffee	
16:00–16:30	**Vendor Session 1** / **Vendor Session 2**		Invited Talk:	
16:30–17:00	Special session: P. Leroy, IBM France *An Invitation to Ada 2005*		S. Vinoski, IONA Technologies, USA *Can Middleware Be Reliable?*	
17:00–17:30	General Assembly & Welcome Cocktail	North Coast Tour & Banquet	Closing Session / Awards	

Tutorial Schedule

Monday June 14th	**T 1**	full day	J.-P. Rosen *Developing a Web server in Ada with AWS*
	T 2	morning	M. Heaney *Programming with the Charles container library*
	T 3	morning	G. Bernat *Probabilistic worst case execution time analysis*
	T 4	afternoon	M. A. Alves *No pointers, great programs*
	T 5	afternoon	A. Strohmeier *Requirements analysis with use cases*
Friday June 18th	**T 6**	full day	P. Amey and R. Chapman *Practical experiences of safety and security critical technologies*
	T 7	full day	B. Lewis and E. Colbert *Developing fault-tolerant, time-critical systems with AADL, UML and Ada*
	T 8	full day	B. Brosgol *Real-Time Java for Ada Programmers*

Lunch is included for those attending full-day programs or two half day tutorials. Additional lunch tickets are on sale throughout the conference.

Tutorials will take place from 9:00 to 12:30 (morning session), and from 14:00 to 17:30 (afternoon session). There will be half an hour coffee breaks at 11:30 and 15:30.

Exhibition

The exhibition opens in mid-morning break on Tuesday and runs until the end of the afternoon break on Thursday. It takes place in a room beside the conference rooms.

The coffee breaks are held in the same exhibition room. The breaks are one hour to allow the attendees a comfortable visit of the exhibition.

Guide to the Social Program

Tuesday Evening: Welcome Cocktail

The city council will offer a welcome cocktail at the beautiful Bellver Castle, a well-preserved fortification built at the late 14th and early 15th centuries to serve both as a fortress and as a residence fot the kings of Mallorca. From the castle, the visitor can enjoy an impressive view of the bay of Palma and the island.

A bus will depart from the hotels to bring the attendees to the castle and the same bus will bring them back.

Wednesday Evening: A tour to the northern coast and the banquet

At 6:00 pm, a bus will depart from the hotels for a sightseeing of the picturesque northern coast of the island. The tour will include a visit to the charming village of Valldemossa.

At 9:00 pm the bus will be back to the hotels, so that the attendees can get ready for the banquet, that will take place at the exquisite restaurant Bahía Mediterráneo, which is very near to the conference site, roughly five minutes walking.

Additional tickets for the banquet are on sale at the conference reception desk.

Registration and Accommodation

Conference Registration:

Three days of conference (June 15th–June 17th) including one copy of the proceedings, coffee breaks, lunches, welcome cocktail on Tuesday 15th, as well as excursion and banquet on Wednesday 16th.

	member Ada-Europe or ACM SIGAda		non member	
	non academia	academia	non academia	academia
Early registration (by May 16th)	530 EUR	470 EUR	590 EUR	530 EUR
Late registration (after May 16th)	590 EUR	590 EUR	650 EUR	650 EUR
Individual day registration (per day)	270 EUR	270 EUR	300 EUR	300 EUR

Tutorial Registration:

The prices are per tutorial, including tutorial notes and coffee breaks. Lunches are only included when registered for a full day tutorial or two half day tutorials.

	half day	full day or two half day
Early registration (by May 16th)	120 EUR	230 EUR
Late registration (after May 16th)	150 EUR	290 EUR

Note: No registration request will be confirmed until the payment has been received. CANCELLATIONS must be in writing. A Cancellation fee of 120 EUR will be applied to all cancellations. No refunds will be given for cancellations postmarked after June 1st. Substitutions will be accepted.

For latest information see the web page at http://www.ada-europe.org/conference2004.html. For additional information, please contact: Albert Llamosí (conference chair), Dept. CC. Matemàtiques i Informàtica, Universitat de les Illes Balears, Carretera de Valldemossa km 7.5, E-07122 Palma, Spain. Tel: +34 971 17 29 67, Fax: +34 971 17 30 03. Email: llamosi@uib.es

Accommodation:

The organization has reserved a limited number of rooms at two hotels, namely: the Palas Atenea (the conference site) and the Mirador, which is roughly 300 m away from the Palas Atenea. The rooms will be allocated on a strict first-come-first-served basis. We strongly advice all participants to book early, as June is a very busy period for hotels in Mallorca.
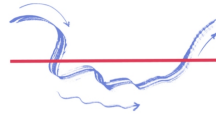
The rates of the hotels per room and per night (including breakfast and VAT) are:

	Single room	Double room
Palas Atenea	135.89 EUR	158.90 EUR
Mirador	100.05 EUR	126.80 EUR

The total amount must be paid directly to the hotel when checking out. Reservations must be guaranteed by means of a credit card (Eurocard/Mastercard-Visa Card) or by a cheque for an amount equivalent to 1 night. In case of cancellation less than 7 days before arrival or "no show" (reservation not cancelled before arrival), 1 night will be charged to your credit card or your cheque will not be given back.

For booking, please send the attached accomodation form to **Barceló Business Travel** either at the fax or at the mail address indicated in the form.

For additional information you can contact this travel agency through the special e-mail address they have for the conference (ada2004@barcelobusiness.com).

REGISTRATION FORM

PARTICIPANT *Please use block capitals*

Ms/Mrs [ ] Mr [ ] Title _____

Family name _____ First name _____

Affiliation/Organization _____

Street _____

City _____ Post/Zip code _____ Country _____

Telephone _____ Fax _____ Email _____

Special requirements (e.g. diet) _____

Reduced registration fee

[ ] member Ada-Europe; national organization _____ [ ] academia

[ ] member ACM; membership number _____

Additional Comments _____

Registration time Early registration (by May 25th) [ ] Late or on site (after May 25th) [ ]

REGISTRATION FEES (see table on previous page)

Conference registration fee

Three day conference . . . . EUR _____

Individual days (Tue [ ] Wed [ ] Thu [ ]) . . . . EUR _____

Tutorial registration Please indicate the tutorials for which you want to register:

Monday, June 14th [ ] T1 [ ] T2 [ ] T3 [ ] T4 [ ] T5

Friday, June 18th [ ] T6 [ ] T7 [ ] T8

Tutorial registration fee . . . . EUR _____

Extra Banquet ticket: _____ tickets @ 64 EUR . . . . EUR _____

Extra proceedings: _____ proceedings @ 30 EUR . . . . EUR _____

TOTAL PAYMENT DUE . . . . EUR _____

PAYMENT METHOD
By credit card. Credit card information:

[ ] Visa [ ] Mastercard

Credit Card Number:
_____

Cardholder Name:
_____

Credit Card Expiration Date:
_____

Signature:
_____

By bank draft to account number: ES97-2051-0151-61-0508477067, of *Ada-Europe 2004*. The account is from the Caixa de Balears (Sa Nostra), E-07120 Palma de Mallorca with bank identifier code / swift code: CECAESMM051 (Please attach a copy of the bank draft to this form).

By cheque, for EUR _____, made payable to *Ada-Europe 2004*.

Mail or fax this form to: Albert Llamosí, Dept. CC. Matemàtiques i Informàtica, Universitat de les Illes Balears, Carretera de Valldemossa km 7.5, E-07122 Palma de Mallorca, Spain. Fax: ++34 971 17 30 03
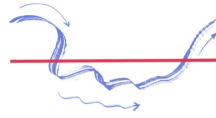
9th International Conference on Reliable Software Technologies - Ada-Europe 2004

Palma de Mallorca, Spain, June 14-18, 2004

ACCOMMODATION FORM

PARTICIPANT   *Please use block capitals*

Ms/Mrs [ ]   Mr [ ]   Title _____

Family name _____ First name _____

Affiliation/Organization _____

Street _____

City _____ Post/Zip code _____ Country _____

Telephone _____ Fax _____ Email _____

BOOKING DATA Please select your choice:

Hotel	[ ] Palas Atenea	[ ] Mirador
Room type	[ ] Single	[ ] Double

Arrival date _____ June 2004   Departure date _____ June 2004   Number of nights _____

BOOKING CONDITIONS

Reservations must be guaranteed by means of a credit card (Eurocard/Mastercard-Visa Card) or a cheque (or a bank transfer) for an amount equivalent to 1 night. In case of cancellation less than 7 days before arrival or "no show" (reservation not cancelled before arrival), 1 night will be charged to your credit card or your cheque (or bank transfer) will not be given back.

The total amount (or the pending amount, if the booking was guaranteed by a cheque or bank transfer) shall be paid directly to the hotel when checking out.

PAYMENT METHOD

Please make cheques or bank drafts payable in EUR. Payments in other currencies will not be accepted.

By credit card. Credit card information:

[ ] Visa   [ ] Mastercard

Credit Card Number: _____

Cardholder Name: _____

Credit Card Expiration Date: _____

Signature: _____

By bank draft to account number: 0182-1048-62-0201507666, of *Barceló Business Travel*. The account is from the Banco de Bilbao Vizcaya Argentaria (BBVA), swift code: BBVAESMMXXX. (Please attach a copy of the bank draft to this form).

By cheque, for EUR _____, made payable to *Barceló Business Travel*.

Mail or fax this form to: Barceló Business Travel, Via Augusta 261-263, E-08017 Barcelona, Spain. Fax: ++34 93-2054164
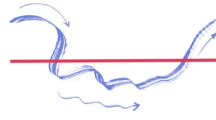
ORGANIZATION

Conference Chair

Albert Llamosí
Dept. of Mathematics and
Computer Science
University of the Balearic
Islands (UIB)
E-07122 Palma, Spain
llamosi@uib.es

Program Co-Chairs

Albert Llamosí
Dept. of Mathematics and
Computer Science
University of the Balearic
Islands (UIB)
E-07122 Palma, Spain
llamosi@uib.es

Alfred Strohmeier
Swiss Fed. Inst. of Technology
in Lausanne
Software Engineering Lab
CH-1015 Lausanne EPFL,
Switzerland
Alfred.Strohmeier@epfl.ch

Tutorial Chair

Javier Miranda
Applied Microelectronics Re-
search Institute
University of Las Palmas de Gran
Canaria
E-35003 Las Palmas, Spain
jmiranda@iuma.ulpgc.es

Exhibition Chair

Peter Dencker
Aonix GmbH
D-76137 Karlsruhe, Deutschland
dencker@aonix.de

Publicity Chair

Dirk Craeynest
Aubay Belgium & K.U. Leuven
B-1180 Brussels, Belgium
Dirk.Craeynest@cs.kuleuven.ac.be

Local Organization Chair

Gabriel Fontanet
Dept. of Mathematics and
Computer Science
University of the Balearic
Islands (UIB)
E-07122 Palma, Spain
gfontanet@uib.es

Miquel Mascaró
Dept. of Mathematics and
Computer Science
University of the Balearic
Islands (UIB)
E-07122 Palma, Spain
mascport@uib.es

**Ada-Europe Conference
Liaison**

Laurent Pautet
Ada-Europe Conference Liaison
ENST Paris
F-75013 Paris, France
pautet@enst.fr

In cooperation with

SIGAda
(Approval pending)

The organizers thank the supporters of the conference