Reliable Software Technologies – Ada-Europe ’96

1996 Ada-Europe International Conference on Reliable Software Technologies
Montreux, Switzerland, June 10-14, 1996 Proceedings
Foreword

The international conference of Ada–Europe, the European federation of national Ada societies, on Reliable Software Technologies took place this year in Montreux, Switzerland, from June 10 to 14, 1996. The conference was organized by Prof. Alfred Strohmeier and Stéphane Barbey on behalf of Ada–Europe and in cooperation with ACM SIGAda.

The conference provides an international forum for researchers, developers, and users of reliable software technologies to share results of research and report on experiences. An important goal is to bring together researchers from academia and practitioners from industry. This year’s conference comprised a three-day technical program and exhibition, surrounded by two days of workshops and tutorials. The exhibition showcased the latest products related to technologies for reliable software systems, including the Ada language.

There is a fourteen-year long tradition of successful Ada–Europe conferences. In the past two years, they were organized together with Eurospace, an organization which groups the major space companies in Europe. Although these two events were quite successful, it was recognized that the interests of the audiences of the two organizations were quite different, and that they could be better served by separate conferences.

It is well known and has been often experienced that quality cannot be added to software as a mere afterthought. This is also true for reliability. Furthermore, reliability of a system is not due to and cannot be built upon a single technology. A wide range of approaches is needed, the most difficult issue being their purposeful integration. Goals of reliability must be precisely defined and included in the requirements, the development process must be controlled to achieve these goals, and sound development methods must be used to fulfill these non-functional requirements.

All artifacts produced must be verified. Useful verification techniques are numerous and complementary: reviewing design documents, proving properties of a program, including its correctness, reasoning about a program, performing static analysis, but also dynamic testing based on program execution, to mention just a few.

Clearly, no assessment of theories and no improvements to practice are possible without quantitative measurement and subsequent statistical interpretation, be it during development, e.g. by counting the number of errors found during reviews, or be it during operation, e.g. by recording the occurrences of faults.

Development of software needs tools, and some are more helpful than others for tracking down errors. Some techniques are well established, such as strong type checking of the source code by the language compiler. Here, the Ada programming language deserves a special mention for it was designed with reliability as a goal. Other techniques are less common and considered as more advanced, such as fault tolerance by replicas in distributed systems.

Clearly, the domain is vast and not all issues related to Reliable Software Technologies can be covered in a single conference, but we are proud to say that these proceedings span a wide range of them and constitute a rich collection of contributions.
This year the conference presented four distinguished speakers, who delivered state-of-the-art information on topics of great importance, for now and for the future:

Programming the Internet in Ada 95  
*S. Tucker Taft*, Intermetrics chief scientist and lead designer of Ada 95, USA

Reliability Modeling for Safety Critical Software  
*Norman E. Schneidewind*, Professor of Information Sciences, Naval Postgraduate School, Monterey, USA

Fault-Tolerance by Replication in Distributed Systems  
*André Schiper*, Professor of Computer Science at the Swiss Federal Institute of Technology in Lausanne, Switzerland

Ada 95: An Effective Concurrent Programming Language  
*Alan Burns*, Professor of Real-Time Systems in the Department of Computer Science, University of York, U.K.

We are very proud to have gained these keynote speakers, and very grateful to them for having authored full papers for inclusion in the proceedings.

This year the number of submitted papers has increased substantially. The program committee selected 35 papers from all around the world, from academia and industry, for inclusion in the proceedings, covering a broad range of software technologies:

- Software Development Methods,
- Verification and Validation,
- Safety and Security,
- Distributed Systems,
- Real-Time Systems,
- Compilers and Tools,
- The Ada 95 Programming Language,
- Interfacing with Other Worlds,
- and Experience Reports.

The conference also comprised a rich choice of tutorials, featuring international experts who presented introductory and advanced material on software engineering:

Object Technology Project Management  
*Richard T. Dué* (Thomsen Dué and Associates, Ltd.)

Software Architecture and Iterative Development Process  
*Philippe Kruchten* (Rational Software)

OOP with Ada 95 and other gOODies  
*John Barnes* (JB Informatics)

Writing Java™-Compatible Applets in Ada 95  
*S. Tucker Taft* (Intermetrics)

Information Systems Programming in Ada 95  
*Benjamin M. Brosgol* (Thomson Software Products)

Real-Time and Distributed Features of Ada 95  
*Joyce L. Tokar* (Tartan)

Task Schedulability Analysis  
*Vance Christiaansen* (Cintech Consulting)

Real-Time POSIX  
*Michael Gonzalez Harbour* (EE & CS Dept., Cantabria University)
Many people contributed to the success of the conference. The role of the Program Committee, reviewing the abstracts, selecting the full papers, shepherding some of them, was, of course decisive.

The work done by Stéphane Barbey deserves special mention: he acted as a tutorial chair, prepared wonderful Web pages for the conference, maintained them, and, last but not least, laid out the advance program brochure. I am also grateful to Thomas Wolf who did most of the clerical work for the preparation of this volume.

Special thanks are due to the Swiss National Research Foundation for sponsoring by a grant the invitation of outstanding researchers.

I hope the participants will enjoy the exciting program, including the social events, of the International Conference on Reliable Software Technologies sponsored by Ada-Europe.

March, 1996

Alfred Strohmeier
Program Committee

Program Chair
Alfred Strohmeier, Swiss Federal Institute of Technology in Lausanne

Tutorial Chair
Stéphane Barbey, Swiss Federal Institute of Technology in Lausanne

Program Committee Members
Angel Alvarez, Technical University of Madrid
Lars Asplund, Uppsala University
Mark S. Gerhardt, LORAL Space & Range Systems
Charlene Roberts-Hayden, GTE Systems
Björn Källberg, CelsiusTech Systems AB
Jan van Katwijk, Delft University of Technology
Philippe Kruchten, Rational
Peter E. Obermayer, Competence Center Informatik GmbH
Laurent Pautet, Telecom Paris
Erhard Ploedereder, University of Stuttgart
Jean-Pierre Rosen, ADALOG
Sergey Rybin, Moscow State University
Edmond Schonberg, New York University
Bill Taylor, Transition Technology Limited
Stef Van Vlierberghe, OFFIS N.V./S.A.
Peter Wehrum, Rational
Brian Wichmann, National Physical Laboratory

Advisory Board Members
John Barnes, John Barnes Informatics
Luc Bernard, OFFIS N.V./S.A.
Alan Burns, University of York
Dirk Craeynest, OFFIS N.V./S.A.
Xavier Cusset, CS Defense
Albert Llamosi, Universitat Rovira i Virgili
Karlotto Mangold, ATM Computer GmbH
Jim Moore, MITRE Corp
Kiyoshi Ishihata, Meiji University
# Table of Contents

## Invited Papers

- **Programming the Internet in Ada 95**  
  *S. Tucker Taft*  
  Page 1

- **Reliability Modeling for Safety Critical Software**  
  *Norman F. Schneidewind*  
  Page 17

- **Fault-Tolerance by Replication in Distributed Systems**  
  *Rachid Guerraoui, André Schiper*  
  Page 38

- **Ada 95: An Effective Concurrent Programming Language**  
  *Alan Burns, Andy J. Wellings*  
  Page 58

## Software Development Methods

- **Mapping HRT-HOOD Designs to Ada 95 Hierarchical Libraries**  
  *Juan Antonio de la Puente, Alejandro Alonso, Angel Alvarez*  
  Page 78

- **An Approach to Increasing Software Component Reusability in Ada**  
  *Hyoseob Kim, Cornelia Boldyreff*  
  Page 89

- **Iterative Software Development for Large Ada Programs**  
  *Philippe Kruchten, Christopher J. Thompson*  
  Page 101

- **HCSD Unit Development Process: Step-Wise Process Improvement**  
  *David Emery, Jaswinder S. Madhur*  
  Page 111

## Verification and Validation

- **Testing Ada 95 Programs for Conformance to Rapide Architectures**  
  *Neel Madhav*  
  Page 123

- **Tasking Deadlocks in Ada 95 Programs and Their Detection**  
  *Jingde Cheng, Kazuo Ushijima*  
  Page 135

- **On Some Characterisation Problems of Subdomain Testing**  
  *T. Y. Chen, Y. T. Yu*  
  Page 147

- **A Framework for Testing Object-Oriented Software Using Formal Specifications**  
  *Rohan Fletcher, A.S.M. Sajeev*  
  Page 159

## Safety & Security

- **Ada 95 and Critical Systems: An Analytical Approach**  
  *Dan Craigen, Mark Saaltink, Steve Michell*  
  Page 171

- **Use of a Static Analysis Tool for Safety-Critical Ada Applications: A Critical Assessment**  
  *Alfred Rosskopf*  
  Page 183
Distributed Systems

Secure Communication in Distributed Ada ............................................................... 198
Jörg Kienzle, Thomas Wolf, Alfred Strohmeier

Using Object-Oriented Methods in Ada 95 to Implement Linda .............................. 211
Kristina Lundqvist, Göran Wall

Shared Packages Through Linda ............................................................................... 223
Göran Wall, Kristina Lundqvist

Drago: An Ada Extension to Program Fault–Tolerant Distributed Applications ...... 235
Francisco J. Miranda, Angel Alvarez, Sergio Arévalo, Francisco J. Guerra

Real–Time Systems

The Dining Philosophers in Ada 95 ........................................................................... 247
Benjamin M. Brosgol

Using Ada 95 for Prototyping Real-Time Systems ................................................... 262
Jorge Real, Agustín Espinosa, Alfons Crespo

The GNARL Implementation of POSIX/Ada Signal Services .................................. 275
Dong-Ik Oh, Ted P. Baker, Seung-Jin Moon

Implementing Protected Types on Embedded Targets ............................................... 287
David Mundie, John Fardo, Ed Kuzemchak

Compilers and Tools

ASIS for GNAT: From the Prototype to the Full Implementation ............................. 298
Sergey Rybin, Alfred Strohmeier, Alexey Kuchumov, Vasily Fofanov

Handling Irregular Grammars in Ada ........................................................................ 312
Drasko Šotirovski, Philippe Kruchten

Interprocedural Call Optimization ............................................................................. 319
Tim Birus, Christine Cipriani, Dean Sutherland

The Ada 95 Programming Language

Augmenting Ada 95 with Additional Real–Time Features ........................................ 330
Johann Blieberger, Roland Lieger, Bernd Burgstaller

Beyond Ada 95: The Addition of Persistence and Its Consequences ........................ 342
Michael J. Oudshoorn, Stephen C. Crawley

Extending the Object–Orientedness of Ada 95 .......................................................... 357
Bernd Holzmüller

An Ada 95 View of Some Difficult or Controversial Points in Object–Oriented Programming ................................................................. 370
Patrick de Bondeli

Where Does GOTO Go to? ........................................................................................ 385
Wolfgang Gellerich, Markus Kosiol, Erhard Ploedereder
Interfacing with Other Worlds

Ada/O2 Coupling: A Solution for an Efficient Management of Persistence in Ada 83 ................................................................. 396
Thierry Millan, Pierre Bazex

An Ada 95 Harness for Converting Legacy Fortran Applications ................. 413
Bernd Gliss

Experience Reports

The Funds Management Modernization: Experiences with Developing an Object-Oriented, Client-Server Management Information System in Ada 95 ........ 427
Chad Bremmon

Converting the Part Task Nautical Simulator to Ada 95 ................................ 439
Kor Molenmaker

Visibility Control and Migration of Interfaces in Large Ada Systems ............ 451
Vincent Celier

Ada Tasking and Dynamic Memory: To Use or Not To Use, That's a Question! .... 460
Philippe Waroquiers

Experiences Applying a Practical Architectural Method .................................. 471
David E. Emery, Richard F. Hilliard II, Timothy B. Rice

A Decade of Development and Deployment of Distributed Ada Systems ........ 485
John D. Smart

Planning the Transition to Ada 95 for a Large Real-Time Project .................. 500
Roland Feith, Michael Tonndorf

Author Index ............................................................................................................ 513