

THE EASST NEWSLETTER  
Volume 7  
December 2003



**European Association of Software Science and Technology**



**EASST Board:**

- Dr. Tiziana Margaria (President),  
email : tmargaria@metaframe.de
- Prof. Dr. Hartmut Ehrig (Vice-President, ETAPS 2000 organizer),  
email : ehrig@cs.tu-berlin.de
- Prof. Dr. Herbert Weber (Treasurer),  
email : herbert.weber@isst.fhg.de
- Dr. Julia Padberg (Secretary),  
email : padberg@tzi.de
- Prof. Dr. Marie-Claude Gaudel (representative in the ETAPS steering committee),  
email : Marie-Claude.Gaudel@lri.fr
- Prof. Dr. Egidio Astesiano (representative in the ETAPS steering committee),  
email : astes@disi.unige.it
- Dr. Michel Wermelinger (column editor),  
email : mw@di.fct.unl.pt
- Prof. Dr. Susanne Graf,  
email : Susanne.Graf@imag.fr

**Homepage of EASST:**

<http://www.easst.org>

**Subscription:**

EASST NEWSLETTER is distributed among the members of EASST, the *European Association of Software Science and Technology*. If you are not yet a member of EASST, but you wish to receive the EASST NEWSLETTER, then you are kindly invited to become a member! Note, there are **no** membership fees. The application form can be found on the last page.

Or apply online at <http://www.easst.org>

**Editor:**

Dr. Julia Padberg  
Technische Universität Berlin  
FB 13, Sekr. FR 6-1, Franklinstr. 28/29, D-10587 Berlin

E-mail : padberg@cs.tu-berlin.de  
URL : <http://tfs.cs.tu-berlin.de/~padberg/>  
Tel : +49-30/314-24165  
FAX : +49-30/314-23516

---



**Contents:**

- Welcome ..... J. Padberg
- Distributed State Space Minimization ..... St. Blom, S. Orzan
- Validating OCL Specifications with the USE Tool ..... P. Ziemann, M. Gogolla
- Report on FM 2003 ..... St. Gnesi
- Report on FMICS'03 ..... Th. Arts, W. Fokkink
- The WICSA Call for Papers
- The EWSA Call for Papers
- The SEFM Call for Papers
- The EASST Flyer
- EASST Application Form



**Welcome!**

**Dear EASST Members,**

again I hope to present an interesting and worthwhile Newsletter.

In this volume we present the summaries of the EASST-award winning papers of FMICS'03:

Bisimulation Reduction on Distributed Memory Machines by Stefan Blom, Simona Orzan  
and

Validating OCL Specifications with the USE Tool by Paul Ziemann, Martin Gogolla

Moreover, we have the reports from FMICS as well as from FM 2003, so those who could not attend know what they have missed.

And there are two *Call for Papers* that may be good to know about.

Yours sincerely,

Julia Padberg  
(EASST-Secretary)

PS.: Again, if you have any contribution you would like to have in the forthcoming newsletter please send it to me.



## Distributed State Space Minimization

Stefan Blom and Simona Orzan \*

\*CWI, The Netherlands

Model checking is a resource intensive application and the most common hardware architecture to provide these resources is the cluster of workstations. Therefore, a lot of interest exist in *distributed model checking tools*. In this paper, we focus on the problem of reducing large state spaces modulo strong bisimulation equivalence. Our target architecture is a cluster of workstations whose nodes are connected by a high bandwidth network (Distributed Memory Machine). Typically the number of processors available is much smaller than the size of the LTS to be reduced.

In [1], a straightforward distributed (message-passing) implementation of the “naive” Kanellakis-Smolka partition refinement algorithm was given. We now propose a new optimized implementation. The optimizations are a marking technique for incremental computation of partitions and a setting where communication and computation can proceed in parallel. This new distributed strong bisimulation reduction outperforms its straight forward counterpart in memory use and in most cases also in time. The gain in time comes from the more elaborated treatment of partition refining. The gain in memory use is the merit of two elements. First, the same improved refinement procedure makes sure that the hashtable accommodates less signatures, thus consumes less memory. The second and more important reason is that computation and communication are not separate phases anymore, but they are interleaved, saving this way the memory needed for storing intermediate results.

**The problem.** The *bisimulation minimization* problem is to find the largest strong bisimulation relation on the states of a given labeled transition system  $(S, T, s_0)$ . This is equivalent to finding a partition  $\pi$  of  $S$  (partition = set of disjoint blocks that cover  $S$ ) s.t.:

- for any  $p, q \in B \in \pi$ , any label  $a$  and  $B' \in \pi$  :  

$$\exists p' \in B' : (p, a, p') \in T \iff \exists q' \in B' (q, a, q') \in T$$
- $\pi$  has the fewest blocks.

**The naive approach.** In the solution presented in [1], successive refined partitions (a partition  $\pi'$  is a refinement of a partition  $\pi$  if every block of  $\pi'$  is contained in a block of  $\pi$ ) are computed, until no refinement is possible. In every iteration, the blocks of the next partition contain states that cannot be distinguished w.r.t. the current partition. The distinction is made by states' signatures. A *signature* of a state  $s$  with respect to  $\pi$  is the set of  $s$ 's outgoing transitions to blocks of  $\pi$ :  $sig_\pi(s) = \{(a, B) \mid (s, a, s') \in T \text{ and } s' \in B \in \pi\}$ . In the actual implementation, an unique ID (an integer) is assigned to each block and partitions are represented as arrays of IDs. The naive algorithm computes the signatures of all states and randomly assigns IDs to each signature, thus keeping the states with the same signature in the same block.

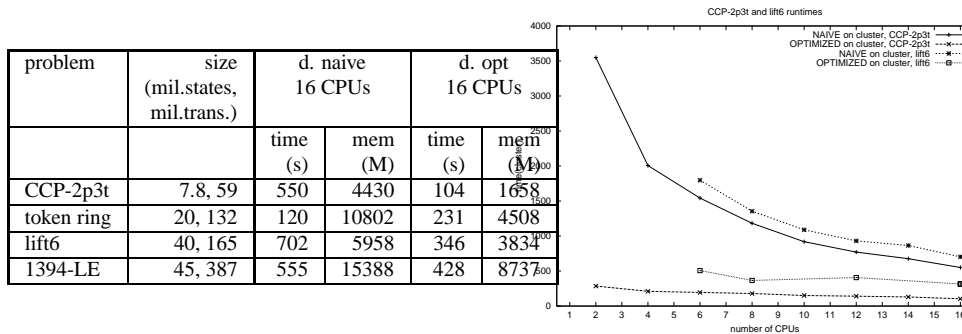


Figure 1: Naive vs. optimized, on 16 processors      Runtimes scaling (*lift6*, *CCP-2p3t*)

**The optimized distributed algorithm.** In the scheme above, *all* signatures are recomputed in *every* iteration, which can be an unnecessary and costly effort. The main idea the “optimized” approach is to mark the states that might have suffered a signature change, i.e. those with an outgoing transition to a state whose ID changed in the current iteration. In the next iteration, only the signatures of the marked states (*unstable*) need to be recomputed. Since the marks are put on states and not on blocks, extra attention has to be paid to ensure the correctness of the splitting procedure, but it pays off: avoiding working on blocks provides parallel/distributed workers with a high(er) degree of independence.

The distributed implementation consists of a series of asynchronous iterations, in between which the  $W$  workers (processes) synchronize in order to decide whether the final partition has been reached. A worker ( $W_i$ ) has two threads: a *segment manager*, that “owns” a subset  $S_i$  of  $S$ , – for each owned state, it stores the incoming transitions, the label and destination ID of the outgoing transitions, and the unstable mark; and a *signatures server*, that maintains a part of a distributed hashtable used to map the signatures w.r.t. the current partition to new IDs (the block identifiers of the next iteration). There are five phases within an iteration:

- (1) managers compute the signatures of their unstable states and send them to the appropriate (i.e., as determined by the hash function) servers
- (2) servers receive the signatures and insert them in their local hashtable
- (3) servers assign new IDs to signatures and send them to the managers
- (4) managers receive the new IDs for their unstable states and send messages to the parent states of the own states that changed the ID
- (5) managers receive and process the update messages

Due to the division of tasks between managers and servers, the first and the second phase happen in parallel, as well as the last three. This limits the amount of CPU idle time, by allowing computation and communication to proceed in parallel. The overlapping of phases also saves memory: since the consumers and producers of messages are active in the same time, the messages don’t have to be stored.

**Experiments.** Some performance comparisons of the new (optimized) and old (naive) distributed implementations are showed in Figure 1. The test set consisted of state spaces generated by case studies carried out with the  $\mu$ CRL toolset. As seen in the table, the runtimes of the optimized implementation outperform the naive one in most cases. For state spaces like the Token Ring protocol, where almost all blocks split in every iteration, and the whole process ends in just a few rounds, the naive works faster,



since it doesn't waste time on administration issues. In all examples though, the memory gain is obvious. The graph shows how the runtimes scale with the number of processors. For both distributed implementations and both case studies, the memory use scales well, i.e. the total memory needed is almost constant.

## References

- [1] Stefan Blom and Simona Orzan. A distributed algorithm for strong bisimulation reduction of state spaces. In Lubos Brim and Orna Grumberg, editors, *PDMC 2002*, volume 68 of *Electronic Notes in Theoretical Computer Science*. Elsevier Science Publishers, 2002.



# Validating OCL Specifications with the USE Tool—An Example Based on the BART Case Study

Paul Ziemann and Martin Gogolla \*

\*Department of Computer Science

University of Bremen, Germany

{ziemann|gogolla}@informatik.uni-bremen.de

## 1 Introduction

The Object Constraint Language (OCL) is part of the Unified Modeling Language (UML). Within software engineering, UML is regarded today as an important step towards development of high-quality object-oriented systems. OCL allows to sharpen UML diagrams through invariants as well as pre- and postconditions.

Tool support for OCL is beginning to develop. Among the first available tools was our system USE (UML Specification Environment). The main task of USE is to validate specifications consisting of UML class diagrams together with OCL invariants and pre- and postconditions that are kept in a USE specification file. By validation, we mean that the developer can give test cases by means of object diagrams and manipulations of them and check whether the USE responses meet the intuition. There are special USE commands for creating and manipulating object diagrams that can be accumulated in command files.

Also part of the USE system is a so-called snapshot generator based on the language ASSL (A Snapshot Sequence Language), which offers more powerful means for state manipulation than the USE commands. It offers loop constructs and other helpful features. With ASSL procedures, the manipulation can be done in a more flexible way.

This paper is a summary of [ZG03] that explains the functionality of USE by means of an example based on the BART case study.

## 2 Specification and Validation with USE

The BART case study description informally describes a portion of the Advanced Automatic Train Control (AATC) system being developed for the Bay Area Rapid Transit (BART) system. BART provides commuter rail service for part of California's San Francisco bay area. The overall objective of the case study is to construct a system within the given infrastructure that can control the speed and acceleration of trains in the system subject to various constraints, the central ones being the following: (1) A train should not enter a closed gate. (2) A train should never get so close to another train in front that if





the train in front stopped suddenly, the (following) train would hit it. (3) A train should stay below the maximum speed that segment of track can handle.

We have created a UML/OCL specification consisting of the following four parts: The base of the specification is a UML class diagram that captures the important elements of the system (train, track etc.), their attributes and relationships. All side effect-free operations of the classes in the diagram are specified by OCL expressions representing the result of the operations. The class diagram is supplemented with a list of OCL invariants (part 1).

Then, we have three invariants representing the three safety requirements mentioned above (part 2). The physical behavior of trains is specified by an ASSL procedure that lets time elapse half a second and updates the positions and speeds of the trains accordingly (part 3). Finally, we created an ASSL procedure that controls a train by setting a new “commanded speed” and “commanded acceleration” of it (part 4). The train will travel with this acceleration until the speed is reached. Part 4 can be seen as an implementation that fulfills the safety requirements (part 2) in the BART environment (part 1) if the physical behavior is as specified (part 3). This implementation can be executed in USE in order to validate the specification.

To validate the specification, we created four scenarios. Each of them contains the same sample track taken from the original case study description. The scenarios differ in the configuration of trains (i.e., in number, position, destination of trains etc.) or state of gates (open or closed). The sequence of actions performed for a test run is as follows: (1) load specification file, (2) create initial system state of scenario, (3) control trains, (4) move trains, (5) repeat steps 3 and 4 until all trains have stopped. When a constraint of the specification is violated, the evaluation browser of the USE tool could be used to locate the error in the implementation.

### 3 Conclusion

The validation exposed numerous faults in our specification/implementation. Some of them have been adopted from the original case study description, such as gaps in the sample track. Others were simple syntax errors that could be fixed quickly, but the validation also helped us to find major errors in our move and control algorithms.

### References

- [ZG03] Paul Ziemann and Martin Gogolla. Validating OCL Specifications with the USE Tool—An Example Based on the BART Case Study. In Thomas Arts and Wan Fokkink, editors, *Eighth International Workshop on Formal Methods for Industrial Critical Systems, Rros, Norway, June 5–7, 2003*, pages 155–167, 2003.



## The 12<sup>th</sup> International FME Symposium

Pisa, Italy – September 8-13, 2003

<http://fme03.isti.cnr.it> - [fme03@isti.cnr.it](mailto:fme03@isti.cnr.it)

**The 12<sup>th</sup> International FME Symposium FM 2003**, has taken place in Pisa, Italy, from the 8th to the 13th of September 2003 at the CNR Research Area premises.

FM03 is the twelfth in a series of symposia organised by *Formal Methods Europe (FME)*, an independent association whose purpose is to promote and support the industrial use of formal methods for computer systems development.

Pervious FME symposia have been notably successful in bringing together an international and varied community of users, researchers and developers of precise mathematical methods for software development. The scope of **FM 2003** has included all aspects of the use of formal methods for software development especially applied to new application areas that are critical in the new aspects of information society. The symposium has attempted to cover a wide range of themes starting from fundamental theories to practical experiences.

**FM 2003** has been hosted by the CNR institute ISTI (Institute of Science and Technology of Information). **General chair** of the symposium has been **Stefania Gnes** (ISTI-CNR) and the **Program Co-chairs** have been **Keijiro Araki** (Kyushu University) and **Dino Mandrioli** (Politecnico di Milano) with **Alessandro Fantechi**, as **Organizing Committee Chair**. 144 papers were submitted from all continents and more than 25 countries. Submitted papers were both of theoretical and applicative nature, and overall of high quality. 44 were selected according to traditional high standards.



In addition to the presentations of the accepted papers, the symposium has offered talks of the four invited speakers, Kouichi Kishida, Brian Randell, Gerard Holzmann, Jean-Raymond Abrial, two scientific talks by industrial sponsors, 8 tutorials (**Tutorial Chair Mieke Massink**), 7 workshops (**Workshop Chair Tommaso Bolognesi**), and tool demonstrations (**Tool Exhibition Chair Tiziana Margaria**). More than 200 participants have attended to FM03 (see for all the details <http://fme03.isti.cnr.it/>) making it a success.

A special full-day meeting Industrial Day (I-Day) organized by Dines Bjørner and sponsored by FME and CoLogNet was held, the day immediately preceding the FM2003 symposium, devoted to formal methods and industry. This innovation of including an Industrial Day as an important part of the symposium shows the strong commitment to involve industrial people more and more within the Formal methods community. The Industrial Day had two purposes: to focus on industrial use of formal methods within FM2003, and to be the inaugural meeting of a new association, ForTIA, the Formal Techniques Industrial Association. Anthony Hall from Praxis Critical Systems had agreed to be the first chairman of ForTIA.



## 8th Workshop on Formal Methods for Industrial Critical Systems

**Thomas Arts and Wan Fokkink**  
(Program Chairs)

The 8th Workshop on Formal Methods for Industrial Critical Systems (FMICS'03) was co-located with the ERCIM meeting in Røros in Norway, June 5-7, 2003. The FMICS workshops aim to strengthen research on formal methods, and to promote the use of formal methods within industry. This year the workshop attracted over thirty participants.

The fourteen regular contributions to the workshop, which were selected from 25 submissions, dealt in large part with the application of formal methods in industrial case studies. Some notable application areas were smart cards, Java, MPEG and avionic systems. Two invited speakers were present: Reiner Hähnle from Chalmers University of Technology presented a paper entitled *Integration of Informal and Formal Development of Object-Oriented Safety-Critical Software: A Case Study with the KeY System*, and Werner Damm from the University of Oldenburg talked about *Recent Trends in the Development of Safety-Critical Systems*.

The FMICS workshop series is organised by the ERCIM Working Group on Formal Methods for Industrial Critical Systems. On the occasion of the co-location with the ERCIM spring meetings, ten representatives from ERCIM institutes within this Working Group were invited to present an overview of current research on formal methods in their own research groups.

As a gesture of involvement, the European Association for Software Science and Technology handed out two best paper awards, one being given to Paul Ziemann and Martin Gogolla from the University of Bremen, and one to Stefan Blom and Simona Orzan from CWI.





The FMICS Working Group has managed to achieve a broad public visibility. This good interaction with the wider scientific community was recognised by the ERCIM board of directors, and as a result FMICS was, during the workshop dinner, rewarded with the ERCIM award for the most successful Working Group of 2002.

Link:

FMICS Working Group:

<http://www.inrialpes.fr/vasy/fmics/index.html>



<http://wicsa4.cs.rug.nl>

Co-located with ECOOP 2004

Oslo, Norway, 12-15 June 2004

WICSA 4 is the fourth international conference devoted entirely to Software Architecture. Its purpose is to bring together software engineering practitioners and researchers from industry and academia to exchange experiences, results and ideas related to all aspects of software architecture. Its mission is to strengthen and expand its role as the premier conference on architectural issues in software system design, development and maintenance, practitioners as well as academics. We are currently soliciting papers, proposals for sessions, and tutorials from academia and industry.

The theme for WICSA-4 is the role of software architecture in the software lifecycle. With the increasing importance of software architecture, organizations realize that just designing a software architecture early in the development process is not sufficient. The software architecture needs to be used as the basis for development and evolution in all phases. This requires us to consider architecture representation during all phases, as well as means to assure architecture conformance of the overall system.

Topics of interest for WICSA-4 include but are not restricted to:

**Architecture in Industrial Software Engineering**

- Case Studies
- Industrial Best Practices
- Role of the Architect
- Interoperability and Integration
- Training, Education and Certification of Software Architects
- Cultural, Economical and Managerial Aspects

**Software Architecture Techniques**

- Architecture Description Languages
- Methods for Architecture Analysis, Evaluation and Assessment

- Architecture Transformations
- Reverse-engineering software architecture
- Traceability from requirements to software architecture to implementation

**Architecture-centric Reuse**

- Product-line Architectures
- System Architecture/Software Architecture
- Architectural Patterns and Styles
- Domain Specific and Reference Architectures
- Domain Analysis and Engineering
- Component Based Software Engineering



Papers should describe original and significant work on the research and practice of software and system architectures. Papers are limited to 10 proceedings pages (IEEE format). Papers must not have been previously published nor have been submitted to, or be in consideration for, any journal, book, or other conference. All papers should explicitly state the goals of systems or approaches described, discuss relationships to previous work and should use accepted, standard terminology.

### **Tutorials & Workshops**

Please send suggestions for tutorials and workshops to WICSA4 @ cs.rug.nl before the closing date of submissions.

Important dates

Deadline for abstracts and papers: February 16, 2004

Accept/reject notification: April 8, 2004

Deadline final paper submission: April 30, 2004

#### **General chair:**

Jan Bosch, University of Groningen, the Netherlands

#### **Program co-chairs:**

Jeff Magee, Imperial College London, UK

Clemens Szyperski, Microsoft Research, USA

#### **Workshop chair:**

Christine Hofmeister, Lehigh University, USA

#### **Tutorial chair:**

Michael Stal, Siemens, Germany

#### **Publicity chair:**

Frank van der Linden, Philips Research, The Netherlands

#### **Local organization:**

Arne Berre, Sintef, Norway

#### **Web chair:**

Eelke Folmer, University of Groningen, the Netherlands

#### **Program Committee:**

Jonathan Aldrich, Carnegie Mellon University, USA

Don Batory, University of Texas at Austin, USA

Marco Bernardo, University of Urbino, Italy

Paul Clements, Software Engineering Institute, USA

Ivica Crnkovic, Mälardalen University, Sweden

David Garlan, Carnegie Mellon University, USA

Hassan Gomaa, George Mason University, USA

Christine Hofmeister, Lehigh University, USA

Andre van der Hoek, University of California, Irvine, USA

Valerie Issarny, INRIA, France

Paola Iverardi, University of L'Aquila, Italy

Rick Kazman, Software Engineering Institute, USA

Nenad Medvidovic, University of Southern California, USA

Jeff Kramer, Imperial College London, UK

Amy Murphy, University of Rochester, USA

Rob van Ommering, Philips Research, The Netherlands

Dewayne Perry, University of Texas, USA

Mary Shaw, Carnegie Mellon University, USA

Heinz Schmidt, Monash University, USA

Judy Stafford, Tufts University, USA

Michael Stal, Siemens, Germany

Will Tracz, Lockheed Martin Mission Systems, USA

Hans van Vliet, Vrije Universiteit, The Netherlands

Michel Wermelinger, Universidade de Nova de Lisboa, Portugal

David Wile, Teknowledge Corp, USA

Alexander L. Wolf, University of Colorado, USA



# FIRST EUROPEAN WORKSHOP ON SOFTWARE ARCHITECTURE (EWSA 2004)

Languages, Styles, Models, Tools, and Applications  
(co-located with ICSE 2004)  
St Andrews, Scotland  
May 21-22, 2004  
<http://www.arch-ware.org/ewsa/2004/>  
e-mail: [ewsa@esia.univ-savoie.fr](mailto:ewsa@esia.univ-savoie.fr)

## IMPORTANT DATES

- Paper submission due: January 9, 2004
- Notification of acceptance: February 9, 2004
- Camera-ready paper due: March 5, 2004
- Workshop: May 21-22, 2004

## SCOPE

The role of software architecture in the engineering of software-intensive applications has become more and more important and widespread. Dynamic software architectures are key to the design, development and evolution of large applications.

This workshop focuses on architecture description languages, architectural styles, architectural models, and architecture-centric tools for modeling, analyzing, refining, building, and monitoring software applications. In particular, the workshop will concentrate on architecture-centric formalisms, technologies, and processes for engineering applications that are dynamic, adaptive, and/or evolvable.

The purpose of the workshop is to bring together researchers and practitioners from academia and industry who are interested in software architecture technology. It addresses both practical and theoretical advances.

## TOPICS

Topics of interest include (but are not limited to):

- semi-formal and formal architecture description languages,
- architectural styles, domain specific architectural models,
- architecture analysis, animation, verification, model checking,





- architecture refinement, architecture transformation,
- architecture-based synthesis, component generation and composition,
- technology of components and component-based frameworks,
- architecture conformance, run-time monitoring,
- architecture reengineering, recovery,
- architecture-based support for reconfigurable, dynamic, adaptive, and/or evolvable applications,
- software tools and environments for architecture-centric software engineering,
- software process models and frameworks for architecture-centric software engineering,
- architecture-centric middleware,
- architectural styles and models for applications based on mature and emerging technologies (Web Services, CORBA, Java/J2EE, .Net, ),
- architecture of distributed applications on wide area networks (web, e-commerce, m-commerce, ),
- industrial applications, case studies, best practices, and experience reports on software architecture,
- other aspects and applications related to software architecture.

## **TYPES OF PAPERS**

We seek three types of papers:

- Position papers: which present concise arguments about a topic of software architecture research or practice (in less than 2000 words). Position papers should not be incomplete versions of full papers.
- Full papers: which describe authors' novel research work (motivated, presented and evaluated in less than 6000 words). Full papers must be original contributions, not published, accepted or submitted for publication elsewhere.
- Industrial reports: which describe real-world experiences related to software architectures (less than 6000 words; short papers are also welcome).

The program committee will select a subset of accepted papers for different kinds of presentations at different workshop sessions.

## **PROCEEDINGS**

The proceedings will be published by Springer-Verlag as part of their Lecture Notes in Computer Science series (pending) (<http://www.springer.de/comp/lncs/index.html>).



## PAPER SUBMISSION

The workshop is open to all researchers, developers and users who are involved with or have an interest in software architecture. All prospective participants should submit a position paper, a full paper or an industrial report. The submissions should explain the contribution to the field and the novelty of the work, making clear the current status of the work. Submissions should be sent electronically in PDF, PostScript or Word/RTF, by the submission date, to [ewsa@esia.univ-savoie.fr](mailto:ewsa@esia.univ-savoie.fr).

## ATTENDANCE

Attendance will be limited to about 40 people. Invitation is based on paper submission. The workshop language is English.

## PROGRAM COMMITTEE

**Organizing Chair:** Ron Morrison, University of St Andrews, UK

**Program Co-chairs:** Flavio Oquendo, University of Savoie at Annecy, France  
Brian Warboys, University of Manchester, UK

### Program Committee:

- Ilham Alloui, University of Savoie at Annecy, France
- Dharini Balasubramaniam, University of St Andrews, UK
- Bob Balzer, Teknowledge Corporation, USA
- Jan Bosch, University of Groningen, Netherlands
- Harald Gall, Technical University of Vienna, Austria
- David Garlan, Carnegie Mellon University, USA
- Mark Greenwood, University of Manchester, UK
- Valrie Issarny, INRIA Rocquencourt, France
- Volker Gruhn, University of Leipzig, Germany
- Philippe Kruchten, IBM Canada Ltd, Canada
- Nicole Levy, PRiSM
- University of Versailles, France
- Radu Mateescu, INRIA Rhne-Alpes, Grenoble, France
- Carlo Montangero, University of Pisa, Italy



## **VENUE**

The workshop will be held in the John Honey Building, School of Computer Science, University of St Andrews - <http://www.dcs.st-andrews.ac.uk>. St Andrews is the oldest university in Scotland, founded around 1410 and the third oldest in the UK. The town of St Andrews has a population of about 16,000 people and is situated on the east coast of Scotland in the Kingdom of Fife. The town is famous for education (the University) and as the home of golf. St Andrews is about 50 miles (80 kilometers) from Edinburgh Airport and about 10 miles (16 kilometers) from the city of Dundee. Further information on the town can be found at <http://www.saint-andrews.co.uk>.

## **SPONSORSHIP**

This workshop is sponsored by the European R&D Project ARCHWARE: Architecting Evolvable Software (IST-2001-32360) [www.arch-ware.org](http://www.arch-ware.org)

**CALL FOR PAPERS: SEFM 2004**  
**Second IEEE International Conference on**  
**SOFTWARE ENGINEERING AND FORMAL METHODS**  
Beijing, China 26 –30 September 2004  
<http://www.iist.unu.edu/SEFM2004>

The Second IEEE International Conference on Software Engineering and Formal Methods (SEFM 2004) will be held in Beijing, the capital of China, during 28-30 September 2004. Affiliated workshops and tutorials will be organized on 26<sup>th</sup> and 27<sup>th</sup> September 2004, and call for workshops and tutorials will be circulated separately by the Workshop/Tutorial chair.

#### SCOPE AND OBJECTIVES

The aim of the conference is to bring together practitioners and researchers from academia, industry and government to advance the state of the art in formal methods, to scale up their application in software industry and to encourage their integration with practical engineering methods.

#### TOPICS

SEFM 2004 solicits research papers related to, but not limited to, the following principal topics:

- requirement analysis and specification
- software architectures and their description languages
- software specification, validation and verification
- software design and refinement
- models of programs and systems
- object and multi-agent systems
- coordination and feature interaction
- integration of formal and informal methods
- integration of different formal methods
- component-based development
- service-oriented development
- aspect-oriented development
- formal aspects of security and mobility
- model checking and theorem proving
- fault-tolerant, real-time and hybrid systems
- analysis of safety-critical systems
- formal aspects of software evolution and maintenance
- formal methods for testing, re-engineering and reuse
- light-weight formal methods
- prototyping and visualisation
- CASE tools and tool integration
- application to industrial cases
- formal methods for industrial standardisation
- socio-economic implications of the use of formal methods

substantial experience reports/case studies are also welcome.

#### SUBMISSION GUIDELINES

Submissions must not have been published or be concurrently considered for publication elsewhere. All submissions will be judged on the basis of originality, contribution to the field, technical and presentation quality, and relevance to the conference.

The proceedings of the conference will be published by the IEEE Computer Society Press. Papers should be written in English and not exceed 10 pages in IEEE format. Instructions for authors are available at <http://computer.org/cspress/instruct.htm>. Latex formatting macros can be downloaded from the website at <http://pubftp.computer.org/Press/Outgoing/proceedings/>.

Further information and instructions about submissions can be found on the conference website. Authors are strongly encouraged to use this website to submit their papers in electronic form.

#### IMPORTANT DATES

Submission deadline for abstract: **29 March 2004**  
Submission deadline for papers: **05 April 2004**  
Notification of acceptance: **18 June 2004**  
Camera-ready version due: **12 July 2004**  
Tutorials and workshops: **26-27 September 2004**  
SEFM 2004 in Beijing, China: **28-30 September 2004**

#### General Chairs

Jifeng He (UNU/IIST, Macao SAR China)  
Fuqing Yang (Peking University, China)

#### Program Committee Chairs

Jorge R. Cuellar (Siemens Research, Munich, Germany)  
Zhiming Liu (UNU/IIST, Macao SAR China)

#### Publicity Chairs

Gianna Reggio (University of Genova, Italy)  
Hanpin Wang (Peking University, China)

#### Finance Chair

Antonio Cerone (The Univ of Queensland, Brisbane, Australia)

#### Organization Chair

Hanping Wang (Peking University, China)

#### Tutorial and Workshop Chair

Bernhard Aichernig (UNU/IIST, Macao SAR China)

#### Steering Committee

Manfred Broy (Technical Univ Munich, Germany)  
Antonio Cerone (The Univ of Queensland, Brisbane, Australia)  
Geoff Dromey (Griffith University, Brisbane, Australia)  
Mike Hinchey (NASA GSFC, Greenbelt, MD, USA)  
Mathai Joseph (TRDDC, Pune, India)  
Zhiming Liu (UNU/IIST, Macao SAR China)  
Andrea Maggiolo-Schettini (University of Pisa, Pisa, Italy)

#### Program Committee

Egidio Astesiano (University of Genova, Italy)  
David Basin (ETH Zurich, Switzerland)  
Gabriel Baum (National University of La Plata, Argentina)  
Tommaso Bolognesi (CNR-ISTI, Pisa, Italy)  
Marcello Bonsangue (Leiden University, The Netherlands)  
Jonathan Bowen (London South Bank University, UK)  
Frank de Boer (CWI, Amsterdam, The Netherlands)  
David Deharbe (DIMAp/UFRN, Natal, RN, Brazil)  
Jin Song Dong (National University, Singapore)  
José Luiz Fiadeiro (University of Leicester, UK)  
Riccardo Focardi (University of Venezia, Italy)  
Paul Gastin (University of Paris, France)  
Uwe Glaesser (Simon Fraser University, Burnaby, BC Canada)  
Bernhard Gramlich (Vienna University of Technology, Austria)  
Nicolas Halbwachs (Verimag, Grenoble, France)  
Connie Heitmeyer (ITD, Naval Res Lab, Washington, DC, USA)  
Teruo Higashino (Osaka University, Japan)  
Mike Hinchey (NASA GSFC, Greenbelt, MD, USA)  
Joern Janneck (University of California at Berkeley, USA)  
Zhi Jin (Institute of Mathematics, CAS, Beijing, China)  
Ekkart Kindler (University of Paderborn, Germany)  
Padmanabhan Krishnan (Bond University, Gold Coast, Australia)  
Yassine Lakhnech (IMAG, Grenoble, France)  
Kung-Kiu Lau (Manchester University, UK)  
K. Rustan M. Leino (Microsoft Research, Redmond, WA, USA)  
Xiaoshan Li (Macao University, Macao)  
Huimin Lin (Institute of Software, CAS, Beijing, China)  
Jian Lu (Nanjing University, China)  
Tiziana Margaria (University of Dortmund, Germany)  
Ron van der Meyden (UNSW/NICTA, Sydney, Australia)  
Ana Moreira (New University of Lisbon, Portugal)  
Jonathan S. Ostroff (York University, Toronto, Canada)  
Paritosh Pandya (TIFR, , Mumbai, India)  
Anders P. Ravn (Aalborg University, Denmark)  
Augusto Sampaio (Federal Univ of Pernambuco, Recife, Brazil)  
Vladimiro Sassone (University of Sussex, UK)  
Shmuel Tyszerowicz (Tel-Aviv University, Israel)  
Mark Utting (University of Waikato, Hamilton, New Zealand)  
Heike Wehrheim (University of Oldenburg, Germany)  
Martin Wirsing (Ludwig Maximilian University, Munich, Germany)  
Jim Woodcock (University of Kent, UK)  
Wang Yi (Uppsala University, Sweden)  
Gianluigi Zavattaro (University of Bologna, Italy)  
Naixiao Zhang (Peking University, China)  
John Zic (Motorola Australia, Sydney, Australia)



---

## **European Association of Software Science and Technology EASST**

### **Who are we?**

EASST is a European non-profit Association that aims at promoting research, development and applications in the area of systematic and rigorous engineering of software and systems.

### **What are our aims?**

Software and Systems Engineering does not receive the public recognition it deserves as one of the most advanced technologies with a great impact on Europe's economic and societal prosperity. This is due to a large extent to the low degree of visibility of the community. Especially research is scattered around a rather large number of communities, meetings in different conferences and workshops.

### **How do you benefit?**

When joining us you enter a larger community and you will help to strengthen a new association that is aiming at a better visibility and recognition of your work.

When joining us you will benefit from a cross-fertilisation between a number of subcommittees in joint initiatives, meetings and activities.

When joining us you will have easy access to consolidated information collected from scattered sources.

### **How to participate?**

All information will be made easily accessible by a number of electronic services.



Membership is for free.

Visit our Web-Site: <http://www.isst.fhg.de>

## **Statute of EASST**

### **Name**

European Association of Software Science and Technology

### **Location**

The Association is located in Berlin/Germany.

### **Legal Status**

The Association is a non-profit organization under German law (»gemeinnütziger eingetragener Verein«).

### **Purpose and Nature of Activities**

The purpose of EASST is to promote the development of science and engineering on software intensive-systems, that play an increasing role in Europe's way into the information society. It therefore supports education and qualification in software science and engineering, advises decision makers on appropriate measures, and informs the general public on the impact of technology developments.

### **The Association will**

1. organize the exchange of information and spread research results by appropriate means to the community
2. provide help in the coordination of initiatives and projects in the area
3. organize and/or sponsor conferences like ETAPS and other professional meetings
4. coordinate its activities with other professional associations with the goal to give birth to a joint European association in informatics.

### **Membership**

Ordinary membership in the association is open to individuals and legal entities, including other professional associations that support the goals of the EASST.

Associated membership may be obtained by members of other professional societies after proper agreement between them and EASST. Membership applications are requested in written form as determined by the board.



### **Membership Fee**

A membership fee is not collected initially and may be collected later on, only after a decision taken by the membership at large.

### **Termination of Membership**

Membership may be terminated by the member's resignation.

Membership will be terminated if the interest of the member in the membership in EASST vanishes. Indication of lost interest is abstention from decisions taken in EASST in electronic ballots for more than four times consecutively.

Membership will also be terminated if a membership fee due according to decision taken by the membership at large is not paid after its invoicing and after a second request.

### **Organs and Officers**

#### **General Assembly**

The membership at large constitutes the general assembly of EASST. The General Assembly elects members of the board of EASST once every two years.

The General Assembly meets at least once a year to receive the annual report of the board including a financial and an activities report. An acceptance vote is expected four weeks after the issue of the report.

The General Assembly votes on the statutes of EASST not later than one year after its constitution, and on further amendments to the statute as well as on the dissolution of the association.

#### **Board**

The Board consists of the president, the vice president, the treasurer, the secretary, and four other board members without a particular portfolio.

#### **Voting**

Voting takes place in written form as determined by the board. The acceptance/rejection of the statutes, the amendment of the statute and the dissolution of the association require a two thirds majority of the members taking part in the vote.

#### **Termination**

In the event of the dissolution of the Association any remaining fund shall be disposed of in a manner determined by the General Assembly so as to support the purposes of EASST.



## Application Form

I wish to become a member of EASST.

Please complete the following:

Name, First Name \_\_\_\_\_

Title \_\_\_\_\_

Company/University \_\_\_\_\_

Position \_\_\_\_\_

Street \_\_\_\_\_

Postal Code, City \_\_\_\_\_

Phone \_\_\_\_\_

Fax \_\_\_\_\_

E-Mail \_\_\_\_\_

Date and Signature \_\_\_\_\_

and return this form as soon as possible to:

EASST c/o  
Herbert Weber  
Fraunhofer-Institut für Software- und Systemtechnik  
Mollstraße 1  
D-10178 Berlin

Fax: +49 (0) 30/2 43 06-1 99  
E-Mail: herbert.weber@isst.fhg.de