

THE EASST NEWSLETTER  
Volume 4  
August 2002



**European Association of Software Science and Technology**



**EASST Board:**

- Prof. Dr. Herbert Weber (President),  
email : herbert.weber@isst.fhg.de
- Prof. Dr. Hartmut Ehrig (Vice-President, ETAPS 2000 organizer),  
email : ehrig@cs.tu-berlin.de
- Prof. Dr. Peter Pepper (Treasurer),  
email : pepper@cs.tu-berlin.de
- Dr. Julia Padberg (Secretary),  
email : padberg@cs.tu-berlin.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,  
email : mw@di.fct.unl.pt
- Dr. Tiziana Margaria (Tool Column Editor),  
email : tmargaria@metaframe.de

**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
  
- Advanced Vocational Training in the IT-sector:  
Synchronicity of Learning and Working ..... J. Einhaus, S. Grunwald, J. Caumanns
  
- THE COLUMNS
  - THE TOOL COLUMN ..... T. Margaria  
An Overview of CADP 2001 ( by H. Garavel, F. Lang, R. Mateescu)
  
  - THE CONTINUOUS LEARNING COLUMN ..... H. Weber
  
- The General Assembly 2002 ..... H. Ehrig
  
- Report on IDPT 2002 ..... J. Padberg
  
- The ICGT Preliminary Program
  
- The ICGT Call for Participation
  
- The EASST Flyer
  
- EASST Application Form



**Welcome!**

**Dear EASST Members,**

I am happy to announce the first two columns, THE TOOL COLUMN edited by T. Margaria and THE CONTINUOUS LEARNING COLUMN by H. Weber. I am very grateful for the effort they have put into these contributions. The new columns lead to a broader perspective and more interesting contributions and discussions. Further columns are planned, e.g. on software architecture and evolution.

Moreover you find an exciting and provocative contribution on vocational IT education that has been presented during the EASST workshop on (continuous) software engineering education, March 7 and 8 2002.

Advanced Vocational Training in the IT-sector:  
Synchronicity of Learning and Working

J. Einhaus, S. Grunwald, J. Caumanns  
Fraunhofer Institute for Software and Systems Engineering

Although it concerns at the moment only Germany, it is intended to be a European effort as well. The impact for academia will be massively, so this shows clearly the need to continue and strengthen our efforts on IT education. Any further suggestions and comments are most welcome.

Again, there are events you want to know about:  
Reports on the *General Assembly during ETAPS* and on the *Sixth World Conference on Integrated Design & Process Technology, 2002* give an idea what has happened and the preliminary program and the call for participation of the *First International Conference on Graph Transformation* tell what is going to happen.

Yours sincerely,

Julia Padberg  
(EASST-Secretary)

## Advanced Vocational Training in the IT-sector: Synchronicity of Learning and Working

**Johannes Einhaus\*, Stefan Grunwald\* and  
Dr. Jörg Caumanns\***

\* Fraunhofer Institute for Software and  
Systems Engineering, Berlin (Germany)

**Abstract.** *A new model of advanced vocational training to qualify employees in the IT-sector has been developed. Different from traditional curricula, relevant objectives and subject matters are modelled as 'reference projects' to shape the vocational training measures.*

**Keywords:** vocational training, lifelong learning, integration of learning into work processes, learning infrastructures

### Introduction

The market for vocational training courses in the IT-sector is insufficiently structured and offers only a low degree of transparency for competencies and skills of employees. Thus missing quality standards and the lack of possibilities of certificated career paths necessitated an alliance of both representatives of employers and the trade unions ('Bündnis für Arbeit') to initiate a new structure for further vocational training in this field.

The Fraunhofer ISST has been commissioned by the federal Ministry for Education and Research (BMBF) to develop a concept of turning this into practice. Advanced vocational training must take place within the work process through 'learning by doing'. The acquired knowledge is highly process- and project-oriented. These informal learning strategies are common in many work processes and they comply well with the demands of the IT sector. We intend to embed them into an educational concept. Furthermore, this basic concept aims at developing self-learning strategies and impart the fundamentals of information technology. With these capabilities, an IT specialist is able to keep up with the technical progress. Under the title *Further Professional Education based on Work processes* (German APO), the Fraunhofer ISST has developed and successfully tested a concept promoting an innovative approach to integrating learning into work processes, including a procedure to test and certify the required competencies.

Three levels of qualification will be differentiated in future:

- specialist
- operative professional

- and strategic professional.

Further vocational training of operative as well as strategic professionals is regulated by orders of the German Vocational Training Law.

In the private sector, a certification as a specialist takes place according to certification procedures determined by European, branch-overlapping standards. The participant is acknowledged his competencies (norms and criteria) in accordance to the respective IT-specialist profile.

This paper will explain in short terms in what respect the Fraunhofer ISST contributed to shaping the contents of the IT further education system. In addition to that, we want to show which structures the certification of an IT specialist is based upon and how a particular competence certification model was put into action with the Deutsche Telekom AG.

### **Work process-oriented further vocational training in the IT-sector**

The basis of this training model are the didactic-methodical basics for shaping the new IT further education system in Germany. In reaction to the increasing demand of specialists and to the poorly structured further education market in the field of computer, software and telecommunication training, the Federal Institute for Vocational Training (BIBB) established a system of further education profiles. Acknowledging job-related experiences and the strive for equating professional experiences with Bachelor/Master degrees point out the importance of experience-based learning.

The particular relevance of informal learning processes for the IT further education is also conditioned and increased by the dynamics within the IT-sector. Regulated further education is hardly capable of reacting to changes in qualification requirements simultaneously so that professional competencies like performance can only be provided for by informal learning processes. These learning strategies have co-developed with the dynamically changing IT branch and thus boosting the entire branch over the last years.

Yet, informal learning strategies cannot be the sole source of professional competencies. It is rather the complementary character of theoretical, canonised and experience-based knowledge that makes the use of informal learning strategies in regulated further education inevitable. A synthesis of formal and informal learning strategies aims at facilitating rather unplanned and unconscious learning and evolving it into a more conscious, continuous and coherent learning [4]. In the context of vocational training, experience-based learning is primarily focused on the work process. There is a number of models combining working and learning, from learning in actual work processes up to learning with working-oriented simulations. To which extend learning while working is possible depends on the working contents, the scope, communication- and cooperation opportunities and at last on personal dispositions. The objective of the concept was to create beneficial working conditions, to produce occasions for learning (gaps in knowledge), providing individual learning support and objectify experience-based knowledge.

In the year 2000, the Fraunhofer ISST was commissioned by the Federal Ministry of Education and Research (German BMBF) to identify and classify quality standards for job profiles of the new IT further educational system, so-called *reference processes*.

In addition to that, we were commissioned to adapt the already established concept of work process-oriented further education (APO) to the requirements of the IT-sector, to elaborate and to test it in an operational context [2].

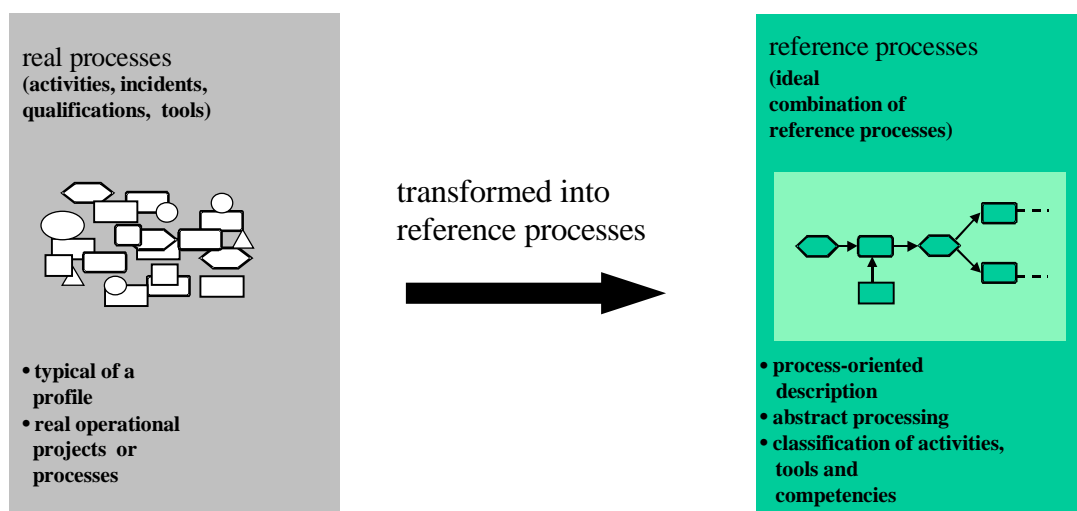
The implementation of new media and modern infra-structures facilitates a learning strategy focused on work place and work processes. Within this concept, work processes make up the reference frame for required competencies and concrete learning objectives.

Those work processes characteristic of a certain job profile are generalised as *reference processes* and function as a major component in the work process-based educational terminology.[3].

Reference processes reproduce genuine work processes in every day IT proceedings and define required competencies for meeting the standards of a certain professional profile.

A reference process thus serves as a frame of reference regarding specific skills and contents of a certain professional profile. The specific skills of a profile are thereby represented in a practical and holistic way as well as derived from an authentic case in an operational sequence.

### Developing reference processes

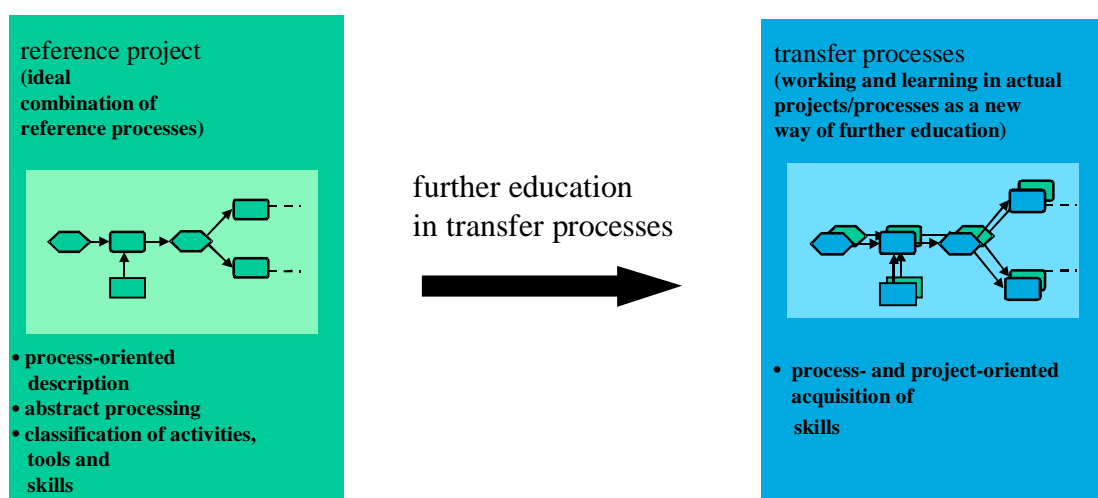


picture 1: from real operational processes to reference processes

The work process-oriented further education then takes place in so-called *transfer projects* where reference processes are translated into the actual operational context (projects, line tasks). They abstractly point out proceedings which enable participants of the project to attain the required skills and at the same time guarantee the sufficiency of this further education model. If, for example, an IT specialist has worked through the reference processes of the job profile „network administrator“ in the respective operational transfer project, he or she will achieve the status of a network administrator.

The concrete learning targets arise from the transfer project and its inherent assessments.

### Qualification standards via reference projects



picture 2: work process-oriented further education in transfer processes by means of reference project

### ACCOMPLISHING WORK PROCESS-ORIENTED FURTHER EDUCATION

Work process-oriented further education means integrating in-house working and learning. The learning aspect of this project is realised in the afore-mentioned transfer processes wherein participants take up operations of the target profile. The tasks that are to be solved cover the proceedings of the reference projects completely. Usually more than one transfer process is necessary to meet the standards of the target job profile.

This further education concept consists of several sequences each containing a specific preparation, task solving and analysis of operational tasks. Besides the actual transaction of reference projects, a well-founded preparation and reflection are of great importance.

Evaluation talks with coaches are meant to make the trainee aware of learning achievements during the work process as well as sorting out previous knowledge and lacks in competence. The implementation of this method has already been put into action successfully in cooperation with the Deutsche Telekom AG. During a time span of 10 months, 11 employees from different Telekom branch offices were trained as network administrators. For this pilot project, the Deutsch Telekom AG and the Fraunhofer ISST were honoured with the Further Education Award by the Federal Institute of Vocational Training (German BIBB).

### **LEARNING RESOURCES**

Even though instruction remains needed in APO oriented vocational training, conventional instructional resources like books, manuals and Computer Based Training become more and more obsolete as contents are getting more unique and very specific. Instructional resources to guide and support the learning process should be such as well which leads to “content on demand” rather than “instruction on demand”. Further more many of the information needed to transfer knowledge from theory into practice is company-specific. Instructional resources that may faster this knowledge transfer must integrate domain specific and company specific „knowledge“.

The appropriate solution to these problems is a flexible network of instructional resources and company-specific documents that can be accessed and integrated on demand. APO makes it easier to set up such an semantic intranet, as the documents created as part of the education can be played back into the “system” as educational resources and knowledge bricks to force the production of new knowledge.

### **Perspectives**

Increasing the transparency on the IT further education market is a necessary objective. The new IT further education system offers certificates with comprehensive standards that can be expected to generate acceptance as a useful orientation tool.

Accordingly, these certificates will in future be highly relevant even in staff decisions.

The constantly growing need for qualified employees as well as an increasing job competition has led to an enormously increased demand for accredited qualification standards with certified degrees. This is the reason for trying to find a solution to the problem of how to adjust national variations of qualification proofs to the changed economic and social conditions on a EU political and professional level (Memorandum on life-long learning, 2000).

The new IT further education system is asked for by business, employers and employees. The high quality of this system and its tight interlocking with the seminal principle of work process-oriented further education will guarantee that this qualification standard will also find a firm ground in the European context.



## References

- [1] Bundesministerium für Bildung und Forschung (BMBF) (Hrsg.) (2002): IT-Weiterbildung mit System. Neue Perspektiven für Fachkräfte und Unternehmen. Dokumentation. Bonn.
- [2] Grunwald, S.; Rohs, M. (2000): Arbeitsprozessorientierung in der IT-Weiterbildung. In: Berufsbildung in Wissenschaft und Praxis 6, 22-24.
- [3] Rohs, M.; Mattauch, W. (2001): Konzeptionelle Grundlagen der arbeitsprozessorientierten Weiterbildung in der IT-Branche. ISST-Bericht 59/01. Fraunhofer-Institut für Software- und Systemtechnik ISST, Berlin.
- [4] Rohs, M. (Hrsg.) (2002): Arbeitsprozessintegriertes Lernen. Neue Ansätze der beruflichen Bildung. Münster u. a.: Waxmann.



## **The Tool Column**

by

Tiziana Margaria

University Dortmund

email : [tmargaria@metaframe.de](mailto:tmargaria@metaframe.de)

As explained in its statute, EASST aims at promoting research, development and applications in the area of systematic and rigorous engineering of software and systems. The fact that this flavour of Software and Systems Engineering does not yet receive the public recognition it deserves - as one of the most advanced technologies with a great impact on economy and society - is partially due to a still insufficient communication style: Citing a senior strategic consultant, "Communication arises at the receiving side". This is a too often ignored truth, and its systems engineering implications are at least threefold: i) there has to be a side that is ready and capable of receiving, ii) communication must happen via a medium that that side can access, and iii) sent messages must make sense to the receiver.

Point i) is a matter of identifying receptive communities. This is addressed by EASST at large, through its diverse activities as a society, and through the activities of its organs and members. This is a hard task, since research is scattered around a rather large number of communities, meeting in different conferences, workshops, and projects, and the potential uptakers of technology are de facto organized in a similar or even more scattered manner.

Point ii) can be optimized by using different media to promote and disseminate content: Conferences, workshops, special contributions, and reports in the newsletter are examples of contributions of different kinds capable of transporting messages of different kind to different people under different circumstances.

Point iii) is a matter of semantics: what content is adequate for which community on which medium? To promote *research, development and applications in the area of systematic and rigorous engineering*



*of software and systems* we must reach the "systems" professionals. We consider tools to be the reification of concepts, and as such the right dissemination medium to reach practitioners who are interested in using the results of research without necessarily accessing research literature on algorithms, complexity, methodologies, and theoretical foundations. Tools are concrete, they allow experimentation, they are the right communication medium if we intend to help application engineers in the field to solve *their* problems without first having to understand *our* (generic) problems and our formulation of a solution.

This issue of the EASST Newsletter hosts for the first time a column on tools. In this column we would like to concentrate on examples of good tools, meaning tools that are useful to solve practical problems and tools that embody good practices of tool design in a particularly effective way. The tool-suite presented in this first column, CADP, is a well established and successful example that fits in both categories. The way this paper is written is specifically aimed at systems engineer: rather than delving in the details of the algorithms, representations, and optimizations, it presents the structure, the purposes, the evolution, and the applications of the CADP tool-suite.

For feedback or proposals of new contributions please contact the column editor Tiziana Magaria.



# An Overview of CADP 2001

**Hubert Garavel and Frédéric Lang and Radu Mateescu**

**INRIA Rhône-Alpes – VASY – 655, avenue de l'Europe, Montbonnot Saint Martin – F-38 334 Saint  
Ismier Cedex, France**

**email: {Hubert.Garavel,Frederic.Lang,Radu.Mateescu}@inrialpes.fr**

**Abstract:** CADP is a toolbox for specifying and verifying asynchronous finite-state systems described using process algebraic languages. It offers a wide range of state-of-the-art functionalities assisting the user throughout the design process: compilation, rapid prototyping, interactive and guided simulation, verification by equivalence/preorder checking and temporal logic model-checking, and test generation. The languages, models, and verification techniques used in CADP have a broad application domain, allowing to deal with communication protocols, distributed systems, embedded software, mobile telephony, asynchronous hardware, cryptography, security, human-computer interaction, etc. CADP is currently used both in industrial companies and academic institutions for research and teaching purposes. During the last years, over 50 applications and case-studies performed using CADP have been reported.

## 1 Introduction

CADP (the CÆSAR/ALDÉBARAN Development Package<sup>1</sup>) is a toolbox for protocol engineering, which offers a wide range of functionalities, from interactive simulation to the most recent formal verification techniques. CADP is dedicated to the efficient compilation, simulation, formal verification, and testing of descriptions written in the ISO language LOTOS [19], a value passing process algebra. It also accepts other input languages such as finite state machines and networks of communicating finite state machines.

July 2001 has seen the release of the new version CADP 2001 “Ottawa”<sup>2</sup>. Among many other features, CADP provides several tools for computing bisimulations (minimizations and comparisons), several model-checkers for various temporal logics and  $\mu$ -calculus, and several verification algorithms including exhaustive verification, on-the-fly verification, symbolic verification using Binary Decision Diagrams, and compositional verification based on refinement. It contains many improvements and five new tools.

The architecture of CADP 2001 is displayed in Figure 1 and explained throughout the paper, which is organized as follows. Section 2 introduces the different languages and models used by the CADP tools. Section 3 describes the main tools of CADP. Section 4 presents the new tools contained in CADP 2001.

<sup>1</sup>Detailed information is available at <http://www.inrialpes.fr/vasy/cadp>.

<sup>2</sup>CADP 2001 “Ottawa” was named in honor of Professor Luigi Logrippo and his research team at the University of Ottawa, who are actively promoting formal methods, especially LOTOS, in the telecommunication industry.



Finally, Section 5 gives concluding remarks.

## 2 Description languages and intermediate models

The CADP toolbox accepts three different input formalisms, materialized by the three grey boxes in Figure 1:

- high-level protocol descriptions written in the ISO language LOTOS [19]: CADP contains two compilers (CÆSAR and CÆSAR.ADT) which translate LOTOS descriptions into C code that can be used for simulation, verification, and testing purposes;
- low-level protocol descriptions specified as Labeled Transition Systems (LTSS, for short), i.e., finite state machines the transitions of which are labeled by action names;
- intermediate-level networks of communicating LTSS, i.e., finite state machines running in parallel and synchronizing together by means of rendezvous; these networks can be expressed in two different formats: EXP (LTSS combined together using LOTOS parallel composition and hiding operators) and FC2 (LTSS combined together using a synchronization product).

The latest releases of the CADP toolbox devote a growing importance to the concept of intermediate formats and programming interfaces. In the sequel of this section, we present the OPEN/CÆSAR environment, which allows the CADP tools to be applied to protocol descriptions written in other languages than LOTOS (e.g.,  $\mu$ CRL, SDL, UML/RT), and the BCG environment, which provides a compact LTS description format together with efficient and useful tools and libraries.

### 2.1 The OPEN/CAESAR environment

OPEN/CÆSAR [13] is an extensible, language-independent Application Programming Interface (API) that allows user-defined programs for simulation, execution, verification (partial, on-the-fly, etc.), and test generation to be developed in a simple and modular way. Various modules have already been written in the OPEN/CÆSAR framework, including: EVALUATOR, an on-the-fly model-checker (Section 4.2), OCIS, an interactive simulator with X-window interface (Section 4.3), TGV, a tool for the generation of conformance test suites based on verification technology (Section 4.5), and many other tools for random execution, deadlock detection, reachability analysis, sequence searching, abstraction of an LTS w.r.t. an interface, etc.

Basically, the OPEN/CÆSAR environment offers primitives to transform a system description into an LTS represented *implicitly* by its initial state and its successor function. Then, every tool connected to the OPEN/CÆSAR environment can take the resulting implicit LTS as input. Three languages have access to the OPEN/CÆSAR environment, namely the BCG graph format (Section 2.2), the LOTOS language, and networks of communicating automata in the EXP format. Since OPEN/CÆSAR is open and well-documented, users can easily extend the environment by adding their own modules or connect their own languages to fit specific needs.

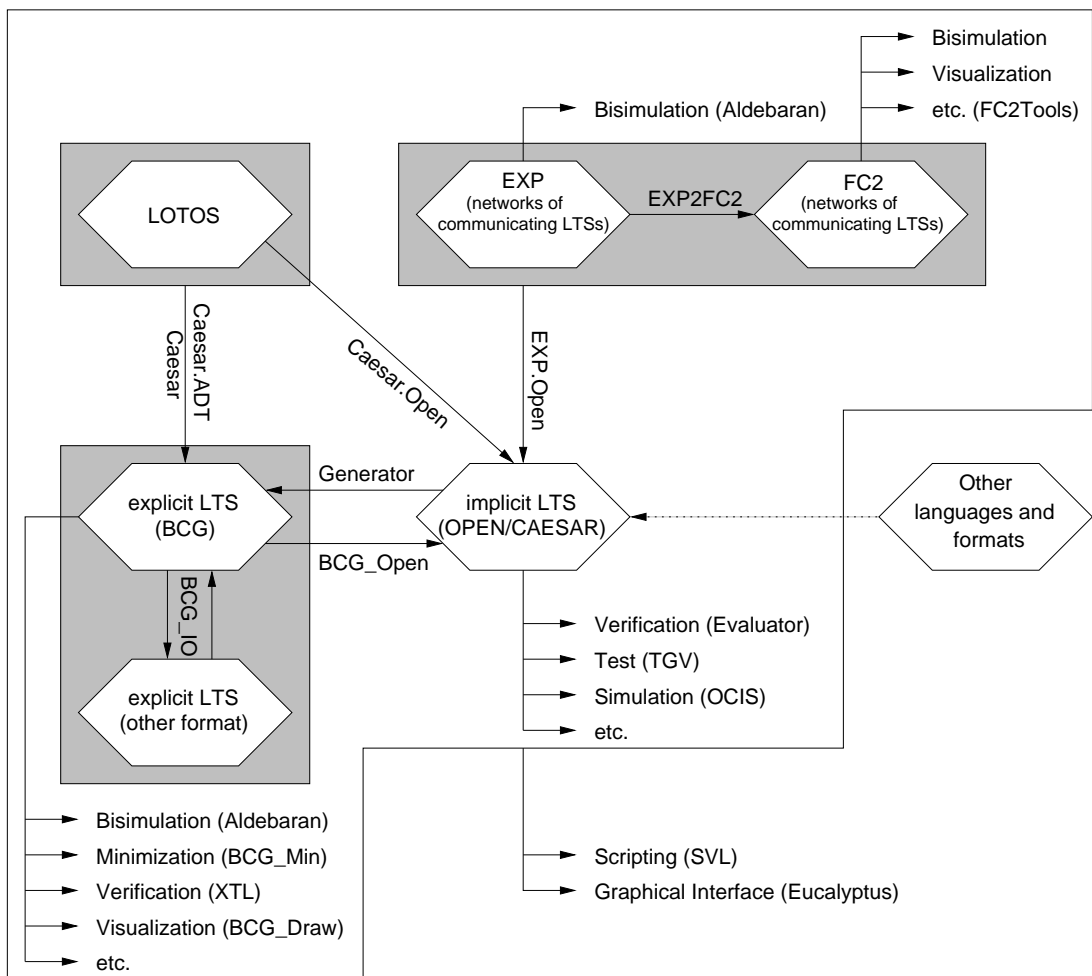


Figure 1: Architecture of CADP 2001 "Ottawa"



## 2.2 The BCG graph format and libraries

BCG (*Binary-Coded Graphs*) [11] is both a format for the representation of explicit LTSSs and a collection of libraries and programs dealing with this format. Compared to ASCII-based formats for LTSSs, the BCG format uses a binary representation with compression techniques resulting in much smaller (up to 20 times) files. BCG is independent from any source language but keeps track of the objects (types, functions, variables) defined in the source programs.

The BCG format supports tools for drawing BCG graphs with an automatic layout of states and transitions, editing the display of BCG graphs interactively, providing information about BCG graphs such as the size of the graph, its number of states and transitions or the list of its labels, performing conversions between the BCG format and a dozen of other formats, hiding and renaming the labels of a graph according to regular expressions, generating dynamic libraries for BCG graphs, minimizing graphs according to strong or branching bisimulation (see the BCG\_MIN tool, Section 4.1), etc.

Simple application programming interfaces are available to read and to produce a BCG graph. Moreover, the BCG\_OPEN tool establishes a gateway between the BCG format (explicit LTSSs) and the OPEN/CÆSAR environment (implicit LTSSs).

## 3 Main tools of CADP

The CADP toolbox contains several tools. In the sequel, we describe the most significant of these tools.

### 3.1 The ALDÉBARAN tool for computing bisimulations

Jointly developed by the VASY team and the VERIMAG laboratory, ALDÉBARAN [6] is a tool for verifying communicating systems, represented by LTSSs. It allows the reduction of LTSSs modulo various equivalence relations (such as strong bisimulation, observational equivalence,  $\tau^* \cdot a$  bisimulation, branching bisimulation, safety equivalence, etc.). It also allows to perform comparison according to strong bisimulation preorder,  $\tau^* \cdot a$  preorder, or safety preorder.

The verification algorithms used in ALDÉBARAN are based either on the Paige-Tarjan algorithm for computing the relational coarsest partition [26], on the “on-the-fly” techniques proposed by Fernandez-Mounier [7], or on symbolic LTS representations using Binary Decision Diagrams (BDDs) [3]. ALDÉBARAN has diagnosis capabilities that provide the user with explanations (counter-example sequences) when two LTSSs are found to be not equivalent.

### 3.2 The CÆSAR compiler

CÆSAR [9] is a compiler that translates the behavioral part of a LOTOS specification into either a C program (to be executed or simulated) or into an LTS.

CÆSAR translation algorithms proceed in several steps. First the LOTOS description is translated into a simplified process algebra called SUBLOTOS. Then an intermediate Petri Net model is generated, which provides a compact, structured and user-readable representation of both the control and data flow. Eventually the LTS is produced by performing reachability analysis on the Petri Net.



CÆSAR accepts full LOTOS with very slight contextual restrictions as regards process recursion. Despite these restrictions, the subset of LOTOS handled by CÆSAR is large and usually sufficient for real-life needs.

The current version of CÆSAR allows the generation of large LTSS (some million states) within a reasonable lapse of time. Moreover, the efficient compiling algorithms of CÆSAR can also be exploited in the framework of the OPEN/CÆSAR environment.

The most recent version of the CÆSAR compiler provides a functionality called EXEC/CÆSAR [15] for C code generation. This C code interfaces with the real world, and can be embedded in applications. This allows rapid prototyping directly from the LOTOS specification.

### 3.3 The CÆSAR.ADT compiler

CÆSAR.ADT [10] is a compiler that translates the data part of LOTOS specifications into libraries of C types and functions. Each LOTOS sort is translated into an equivalent C type and each LOTOS operation is translated into an equivalent C function (or macro-definition). CÆSAR.ADT also generates C functions for comparing and printing abstract data types values, as well as iterators for the sorts of finite domain.

CÆSAR.ADT accepts full LOTOS with the following (quite natural) restriction, as regards the data part: constructor operations must be identified; equations are oriented; there is a decreasing priority between equations; equations between constructors are not allowed. Also, parameterized types are not compiled (yet).

CÆSAR.ADT is fast: translation of large programs (several thousands of lines) is usually achieved in a few seconds. CÆSAR.ADT can be used in conjunction with CÆSAR, but it can also be used separately to compile and execute efficiently large abstract data types descriptions.

### 3.4 The XTL model-checker

XTL (*eXecutable Temporal Language*) [23] is a functional-like programming language designed to allow an easy, compact implementation of various temporal logic operators. These operators are evaluated over an LTS encoded in the BCG format. Besides the usual predefined types (booleans, integers, etc.), the XTL language defines special types, such as sets of states, transitions, and labels of the LTS. It offers primitives to access the informations contained in states and labels, to obtain the initial state, and to compute the successors and predecessors of states and transitions. The temporal operators can be easily implemented using these functions together with recursive user-defined functions working with sets of states and/or transitions of the LTS. A compiler for XTL has been developed, and several temporal logics like HML [18], CTL [4], ACTL [25], and LTAC [28] have been easily implemented in XTL.

### 3.5 The EUCALYPTUS graphical user interface

EUCALYPTUS [12] is a graphical user interface written in Tcl/Tk that integrates the CADP tools in a unified, user-friendly interface. This interface has the name of the project within which it was developed: the Euro-Canadian project "EUCALYPTUS".



Additionally, EUCALYPTUS integrates complementary software such as the APERO data type pre-processor for LOTOS [27], the ELUDO simulator of LOTOS descriptions [31], or the FC2 tools [1], together with the graphical editor AUTOGRAPH [29].

## 4 New tools of CADP 2001

The new release of CADP contains five new tools: BCG\_MIN, EVALUATOR 3.0, OCIS, SVL, and TGV.

### 4.1 The new BCG\_MIN tool for computing bisimulations

Jointly developed by the VASY team and Holger Hermanns (University of Twente), BCG\_MIN implements various minimization algorithms for graphs encoded in the BCG format. It can be used to minimize “standard” LTSS, as well as “probabilistic” and “stochastic” LTSS, which may carry respectively probabilistic and stochastic labels and generalize many theoretical models published in the literature (for instance the Discrete Time Markov Chains and the Continuous Time Markov Chains).

Compared to former LTS minimization tools (including ALDÉBARAN and FC2 tools), BCG\_MIN only implements two equivalences, namely strong and branching bisimulation. For these two equivalences, however, it offers compelling advantages:

- BCG\_MIN can handle larger LTSS, at least larger by an order of magnitude; for instance, the largest LTS reduced so far by BCG\_MIN has more than 7 million states and 40 million transitions; according to Prof. Jan Friso Groote [17], BCG\_MIN is “*the best implementation of the standard (i.e., Groote & Vaandrager [16]) algorithm for branching bisimulation*”;
- BCG\_MIN uses BCG as its native format, thus leading to speed improvement, because BCG occupies much less disk space than most graph formats;
- BCG\_MIN is able to print state equivalence classes in a user-friendly way, by relating the state numbers of the minimized graph to the state numbers of the original graph; in the case of branching equivalence, the  $\tau$ -cycles are properly displayed.

### 4.2 The new EVALUATOR 3.0 model-checker

EVALUATOR 3.0 [24] is a new version of the EVALUATOR tool, which performs on-the-fly verification of regular alternation-free  $\mu$ -calculus formulas on LTSS represented implicitly according to the OPEN/CÆSAR API. Compared to the previous version 2.0 [8], EVALUATOR 3.0 brings major improvements:

- The input specification language of EVALUATOR 3.0 is more powerful than the one of EVALUATOR 2.0:
  - action formulas can contain any combination of boolean operators and basic predicates over transition labels (which can be now given also as UNIX regular expressions over character strings);

- regular transition sequences can be succinctly described using regular formulas built from action formulas and the usual regular expression operators;
  - it is also possible to define macro operators parameterized by formulas and to group them into separate libraries that may be included in the main specification.
- The model-checking algorithm of EVALUATOR 3.0 uses a new on-the-fly boolean resolution algorithm, which has a much better average complexity than the algorithm used in EVALUATOR 2.0. It explores less states before deciding the truth value of the formula, which leads sometimes to dramatic reductions (several orders of magnitude) of the execution time. Moreover, EVALUATOR 3.0 has been optimized in order to work more efficiently when verifying temporal formulas on explicit LTSS encoded as BCG files. Due to these optimizations, the memory consumption and the execution time of EVALUATOR 3.0 have been reduced by up to 5% and 20%, respectively.
  - The diagnostics generated by EVALUATOR 3.0 are improved [22]. Diagnostics are portions of LTSS explaining either the satisfaction or the refutation of a formula: if the formula is false, a diagnostic is a counter-example; if the formula is true, a diagnostic is an example. In particular, the diagnostics obtained for derived “pure” branching-time logics like CTL and ACTL fully explain the semantics of their operators. EVALUATOR 3.0 may also serve to search regular execution sequences in the LTS, by asking for diagnostics of regular modalities.

Three libraries are also available that encode the operators of the ACTL temporal logic as well as a set of generic temporal property patterns defined by Prof. Matthew Dwyer from Kansas State University [5].

### 4.3 The new OCIS interactive graphical simulator

A new interactive, graphical simulator named OCIS (*OPEN/CÆSAR Interactive Simulator*) was added to CADP. Designed to replace the venerable XSIMULATOR, OCIS enables visualization and error detection during the design phase of systems containing parallelism and asynchronous communication between tasks. Its main features are:

- visualization of simulation scenarios as execution traces, trees, or Message Sequence Charts (MSCs),
- manipulation of simulation scenarios, which can be edited, saved as BCG graphs, and loaded again during another simulation session,
- manual (step by step) and automatic (pattern-guided) navigation in the system under simulation,
- source-level debugging, with access to parallel tasks, state variables, etc.
- possibility to modify the source code and to re-compile it without leaving the current simulation session.

OCIS was designed to be as much as possible language-independent and should therefore be usable for any specification language or formalism interfaced with the OPEN/CÆSAR API.



#### 4.4 The new SVL scripting language

A new tool named SVL (*Script Verification Language*) [14] was added to CADP. The SVL language and its associated compiler target at simplifying and automating the verification of LOTOS programs. SVL behaves as a tool-independent coordination language on top of the CADP and FC2 tools, in the same way as EUCALYPTUS is a tool-independent graphical user interface.

SVL offers high-level operators for generation, parallel composition, minimization, label hiding, label renaming, abstraction, comparison, and model-checking of LTSS. It supports several methods of verification (e.g., enumerative, compositional, and on-the-fly), which can be easily combined together.

A compiler for SVL has been developed, which translates an SVL verification scenario into a Bourne shell script, which will perform all the operations needed to execute the verification scenario, e.g., invoking verification tools with appropriate options and parameters, generating intermediate files, etc.

SVL has been used in several case-studies: most of the CADP demo examples (19 demos over a total of 29) take advantage of SVL readability and conciseness. In most cases, SVL allows the user to get rid of Makefiles and shell-scripts as well as many auxiliary files which are generated automatically from a simple SVL script. Because of its expressiveness and robustness, SVL subsumes totally the DES2AUT tool [21] used in previous versions of CADP.

#### 4.5 The new TGV test generator

The latest version of the TGV (*Test Generation based on Verification technology*) [20] tool, jointly developed by the PAMPA team of INRIA Rennes/IRISA and the VERIMAG laboratory, has been integrated into the CADP toolbox. TGV is a tool for the automatic generation of test suites from formal specifications. These test suites are used to assess the conformance of a protocol implementation with respect to the formal specification of this protocol. TGV takes two main inputs:

- a specification of the protocol's behavior, defined as an implicit LTS using the OPEN/CÆSAR API (this API allows TGV to be used for various languages: LOTOS, SDL, UML/RT, etc.),
- a test purpose, which selects the subset of the protocol's behavior to be tested; the test purpose is defined as an explicit LTS, the states of which are either normal states, accepting states (i.e., final states characterizing parts of the protocol satisfying the test purpose), or refusing states (i.e., final states characterizing parts of the behavior that are irrelevant to the test purpose).

To produce conformance test suites automatically, TGV applies algorithms coming from verification technology. Test generation is done “on-the-fly” on the synchronous product of the specification with the test purpose; this product allows to avoid state explosion by exploring only the subset of the protocol specification permitted by the test purpose. The test cases generated by TGV are LTSS, the transitions of which carry test verdicts such as “pass”, “fail”, and “inconclusive”.

## 5 Conclusion

CADP contains a lot of tools and offers a wide variety of functionalities. CADP 2001 “Ottawa” provides several of the best algorithms for simulation and verification. It supports libraries that make the addition



of new tools and the connection to new languages and description formats extremely modular. It also contains a graphical user interface and a scripting language that make its use easier for both expert and non-expert users.

Moreover, CADP is supported, maintained and constantly improved. It is available on LINUX, SOLARIS, and WINDOWS platforms. It is widely distributed: in June 2001, it had been licensed to 239 sites and during year 2000, licenses were granted for 770 machines around the world; from January 1st, to June 26th, 2001, licenses were granted for 797 machines. Additionally, the CADP tools are integrated into the Web-based, open communication platform ETI (Electronic Tool Integration Platform) [30, 2].

During the last years, over 50 applications and case-studies performed using CADP have been published<sup>3</sup> and 10 research tools based upon the OPEN/CÆSAR and BCG environments of CADP have been developed<sup>4</sup>.

## Acknowledgements

We would like to thank all the people who contributed to the development of CADP:

- Moez Cherif, Hubert Garavel, Marc Herbert, Bruno Hondelatte, Pierre Kessler, Frédéric Lang, Stéphane Martin, Radu Mateescu, Aldo Mazilli, Frédéric Perret, Mihaela Sighireanu, and Irina Smarandache of the VASY project at INRIA Rhône-Alpes (Grenoble, France),
- Laurent Mounier and Aline Sénart of the VERIMAG laboratory (Grenoble, France),
- Thierry Jérón, Pierre Morel, and Séverine Simon of the PAMPA project at INRIA/IRISA (Rennes, France),
- Holger Hermanns at the University of Twente (The Netherlands).

We are also extremely grateful to all the scientists who contributed to the development of CADP in the past and who provided us with valuable feedback and advices about the use of CADP.

At last, we thank Solofo Ramangalahy for his useful comments about this paper.

## References

- [1] Amar Bouali, Annie Ressouche, Valérie Roy, and Robert de Simone. The Fc2Tools set: a Toolset for the Verification of Concurrent Systems. In Rajeev Alur and Thomas A. Henzinger, editors, *Proceedings of the 8th Conference on Computer-Aided Verification (New Brunswick, New Jersey, USA)*, volume 1102 of *Lecture Notes in Computer Science*. Springer Verlag, August 1996.
- [2] Volker Braun, Jürgen Kreileder, Tiziana Margaria, and Bernhard Steffen. The ETI Online Service in Action. In Rance Cleaveland, editor, *Proceedings of the 5th International Conference on Tools*

<sup>3</sup>Information is available at <http://www.inrialpes.fr/vasy/cadp/case-studies>.

<sup>4</sup>Information is available at <http://www.inrialpes.fr/vasy/cadp/software>.



- and Algorithms for Construction and Analysis of Systems (TACAS 1999)*, volume 1579 of *Lecture Notes in Computer Science*, pages 439–443, Amsterdam (The Netherlands), 1999.
- [3] R. E. Bryant. Graph-Based Algorithms for Boolean Function Manipulation. *IEEE Transactions on Computers*, C-35(8), 1986.
  - [4] E. Clarke, E. A. Emerson, and A. P. Sistla. Automatic Verification of Finite-State Concurrent Systems using Temporal Logic. In *10th Annual Symposium on Principles of Programming Languages*. ACM, 1983.
  - [5] Matthew B. Dwyer, George S. Avrunin, and James C. Corbett. Patterns in Property Specifications for Finite-State Verification. In *Proceedings of the 21st International Conference on Software Engineering ICSE'99 (Los Angeles, CA, USA)*, May 1999.
  - [6] Jean-Claude Fernandez. An Implementation of an Efficient Algorithm for Bisimulation Equivalence. *Science of Computer Programming*, 13(2–3):219–236, May 1990.
  - [7] Jean-Claude Fernandez and Laurent Mounier. “On the Fly” Verification of Behavioural Equivalences and Preorders. In K. G. Larsen and A. Skou, editors, *Proceedings of the 3rd Workshop on Computer-Aided Verification (Aalborg, Denmark)*, volume 575 of *Lecture Notes in Computer Science*, Berlin, July 1991. Springer Verlag.
  - [8] Jean-Claude Fernandez and Laurent Mounier. A Local Checking Algorithm for Boolean Equation Systems. Rapport SPECTRE 95-07, VERIMAG, Grenoble, March 1995.
  - [9] Hubert Garavel. *Compilation et vérification de programmes LOTOS*. Thèse de Doctorat, Université Joseph Fourier (Grenoble), November 1989.
  - [10] Hubert Garavel. Compilation of LOTOS Abstract Data Types. In Son T. Vuong, editor, *Proceedings of the 2nd International Conference on Formal Description Techniques FORTE'89 (Vancouver B.C., Canada)*, pages 147–162. North-Holland, December 1989.
  - [11] Hubert Garavel. Binary Coded Graphs: Definition of the BCG Format. Rapport SPECTRE C28, Laboratoire de Génie Informatique — Institut IMAG, Grenoble, January 1991.
  - [12] Hubert Garavel. An Overview of the Eucalyptus Toolbox. In Z. Brezocnik and T. Kapus, editors, *Proceedings of the COST 247 International Workshop on Applied Formal Methods in System Design (Maribor, Slovenia)*, pages 76–88. University of Maribor, Slovenia, June 1996.
  - [13] Hubert Garavel. OPEN/CÆSAR: An Open Software Architecture for Verification, Simulation, and Testing. In Bernhard Steffen, editor, *Proceedings of the First International Conference on Tools and Algorithms for the Construction and Analysis of Systems TACAS'98 (Lisbon, Portugal)*, volume 1384 of *Lecture Notes in Computer Science*, pages 68–84, Berlin, March 1998. Springer Verlag. Full version available as INRIA Research Report RR-3352.



- [14] Hubert Garavel and Frédéric Lang. SVL: a Scripting Language for Compositional Verification. In Myungchul Kim, Byoungmoon Chin, Sungwon Kang, and Danhyung Lee, editors, *Proceedings of the 21st IFIP WG 6.1 International Conference on Formal Techniques for Networked and Distributed Systems FORTE'2001 (Cheju Island, Korea)*, pages 377–392. IFIP, Kluwer Academic Publishers, August 2001. Full version available as INRIA Research Report RR-4223.
- [15] Hubert Garavel, César Viho, and Massimo Zendri. System Design of a CC-NUMA Multiprocessor Architecture using Formal Specification, Model-Checking, Co-Simulation, and Test Generation. *Springer International Journal on Software Tools for Technology Transfer (STTT)*, 3(3):314–331, July 2001. Also available as INRIA Research Report RR-4041.
- [16] Jan Friso Groote and Frits Vaandrager. An Efficient Algorithm for Branching Bisimulation and Stuttering Equivalence. In M. S. Patterson, editor, *Proceedings of the 17th ICALP (Warwick)*, volume 443 of *Lecture Notes in Computer Science*, pages 626–638. Springer Verlag, 1990.
- [17] J.F. Groote and J.C. van Pol. State space reduction using partial tau-confluence. Research Report SEN-R0008, CWI, Amsterdam, The Netherlands, 2000.
- [18] M. Hennessy and R. Milner. Algebraic laws for nondeterminism and concurrency. *Journal of the ACM*, 32:137–161, 1985.
- [19] ISO/IEC. LOTOS — A Formal Description Technique Based on the Temporal Ordering of Observational Behaviour. International Standard 8807, International Organization for Standardization — Information Processing Systems — Open Systems Interconnection, Genève, September 1988.
- [20] T. Jérón and P. Morel. Test generation derived from model-checking. In N. Halbwachs and D. Peled, editors, *Proceedings of the Conference on Computer-Aided Verification CAV'99 (Trento, Italy)*, volume 1633 of *Lecture Notes in Computer Science*, pages 108–122. Springer Verlag, July 1999.
- [21] Jean-Pierre Krimm and Laurent Mounier. Compositional State Space Generation from LOTOS Programs. In Ed Brinksma, editor, *Proceedings of TACAS'97 Tools and Algorithms for the Construction and Analysis of Systems (University of Twente, Enschede, The Netherlands)*, volume 1217 of *Lecture Notes in Computer Science*, Berlin, April 1997. Springer Verlag. Extended version with proofs available as Research Report VERIMAG RR97-01.
- [22] Radu Mateescu. Efficient Diagnostic Generation for Boolean Equation Systems. In Susanne Graf and Michael Schwartzbach, editors, *Proceedings of 6th International Conference on Tools and Algorithms for the Construction and Analysis of Systems TACAS'2000 (Berlin, Germany)*, volume 1785 of *Lecture Notes in Computer Science*, pages 251–265. Springer Verlag, March 2000. Full version available as INRIA Research Report RR-3861.
- [23] Radu Mateescu and Hubert Garavel. XTL: A Meta-Language and Tool for Temporal Logic Model-Checking. In Tiziana Margaria, editor, *Proceedings of the International Workshop on Software Tools for Technology Transfer STTT'98 (Aalborg, Denmark)*, pages 33–42. BRICS, July 1998.



- [24] Radu Mateescu and Mihaela Sighireanu. Efficient On-the-Fly Model-Checking for Regular Alternation-Free Mu-Calculus. In Stefania Gnesi, Ina Schieferdecker, and Axel Rennoch, editors, *Proceedings of the 5th International Workshop on Formal Methods for Industrial Critical Systems FMICS'2000 (Berlin, Germany)*, GMD Report 91, pages 65–86, Berlin, April 2000. Also available as INRIA Research Report RR-3899.
- [25] R. De Nicola and F. W. Vaandrager. *Action versus State based Logics for Transition Systems*. In Irène Guessarian, editor, *Semantics of Systems of Concurrent Processes*, volume 469 of *Lecture Notes in Computer Science*, pages 407–419. Springer Verlag, April 1990.
- [26] Robert Paige and Robert E. Tarjan. Three Partition Refinement Algorithms. *SIAM Journal of Computing*, 16(6):973–989, December 1987.
- [27] Charles Pecheur. *Improving the Specification of Data Types in LOTOS*. Doctorate thesis, University of Liège, November 1996. Collection of Publications of the Faculty of Applied Sciences, Nr 171.
- [28] Jean-Pierre Queille and Joseph Sifakis. Fairness and Related Properties in Transition Systems — A Temporal Logic to Deal with Fairness. *Acta Informatica*, 19:195–220, 1983.
- [29] Valérie Roy and Robert de Simone. Auto/Autograph. In R. P. Kurshan and E. M. Clarke, editors, *Proceedings of the 2nd Workshop on Computer-Aided Verification (Rutgers, New Jersey, USA)*, volume 3 of *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, pages 477–491. AMS-ACM, June 1990.
- [30] Bernhard Steffen, Tiziana Margaria, and Volker Braun. The Electronic Tool Integration Platform: Concepts and Design. *Springer International Journal on Software Tools for Technology Transfer (STTT)*, 1–2(1):9–30, December 1997.
- [31] B. Stepien, J. Tourrilhes, and J. Sincennes. ELUDO: The University of Ottawa LOTOS Toolkit. Technical report, University of Ottawa, 1994. Obtainable by FTP on [lotos.csi.uottawa.ca](http://lotos.csi.uottawa.ca).



## **The Continuous Learning Column**

by

Herbert Weber

Fraunhofer ISST

email : [herbert.weber@isst.fhg.de](mailto:herbert.weber@isst.fhg.de)

Dear Colleagues and Friends,

You all probably know how much of a rush followed to the request for "Expressions of Interest" by the commission of the European Union for its 6th Framework Programme. In the meantime things get sorted out at the commission and research on e-teaching/e-learning is one of the hot topics.

Rather little interest has been expressed in harmonization of IT teaching in general independent of whether it happens electronically or not. Since we have started an initiative exactly into that direction we have an interesting USP. This was confirmed in a conversation with Mr. Rodriguez-Rosello who I recently visited in Luxemburg in order to present him our ideas.

It seems we have a good chance to get a piece of the cake if we submit a proposal at the next call in November. For its preparation we will have to sit together again since I got many hints on how we should position ourselves and a proposal. At any rate, we were advised to get in touch with other European initiatives, especially with the industry-consortium "career-space" ([www.cedefop.eu.int](http://www.cedefop.eu.int)).

So, all of you who are interested in the subject should let me know.



## **EASST General Assembly at ETAPS 2002**

**Report by H. Ehrig, April 2002**

Technical University Berlin

The EASST General Assembly in 2002 took place on April 9, during ETAPS 2002 in Grenoble. Due to sudden illness of the EASST President Herbert Weber it was chaired by the EASST Vice-President Hartmut Ehrig. Since the General Assembly was open not only to EASST members, but also to all other ETAPS participants, the report of the vice president started with a short history of EASST, founded in 2000, having about 100 members at the time of ETAPS 2001 in Genova and almost 150 members now. This increase of membership should continue in the following years where all EASST members and especially the EASST representatives of different countries are asked to promote EASST. The membership in EASST is very attractive because of reduced conference fees and several other EASST activities including an EASST Newsletter on one hand and no membership fee on the other hand.

The EASST Newsletter reports about all the activities of EASST and will contain columns on specific topics interesting for EASST. Presently the following columns are confirmed:

- Education Column (Editor: Herbert Weber)
- Tools Column (Editor: Tiziana Margaria).

In 2002 the following papers EASST awards for the best paper in the area of Software Science and Technology have been presented already:

- ETAPS 2002, Grenoble  
*Synthesizing Monitors for Safety Properties*  
by Klaus Havelund and Grigore Rosu
- IDPT (Integrated Design and Process Technology) 2002, Pasadena  
*Specifying Components for NEST Applications*  
by Asuman Sünbül



A further EASST award will be presented during ICGT (International Conference on Graph Transformation) 2002, Barcelona.

In absence of Herbert Weber the EASST Board member Tiziana Margaria reported about the EASST education workshop in March 2002 in Berlin and further actions planned in this direction. A more detailed report on this activity will be given elsewhere.

As announced in the invitation for the General Assembly a new EASST Board member had to be elected, because Jose Fiadeiro resigned from the EASST Board after he had become chairman of the ETQAPS Steering Committee last year. Michel Wermelinger, also from Lisbon, has been nominated by the EASST Board and elected by the EASST General Assembly for the open position.

The next EASST General Assembly will probably take place during ETAPS 2003 in Warsaw.



## **Report on the Sixth World Conference on Integrated Design & Process Technology, Pasadena 2002**

**Julia Padberg\***

\*Institute for Software Engineering and Theoretical Computer Science  
Technical University Berlin

During June, 23 - 28, 2002 the 6<sup>th</sup> World Conference on Integrated Design & Process Technology (IDPT 2002) has taken place, in Pasadena near Los Angeles, California, USA. It has been four days full of intense work and cooperation, enormous amount of interesting and enlightened presentations, lots of deep scientific discussions, and interesting social events.

The IDPT conferences are organized by the Society for Design and Process Science (SDPS). This year it was coorganized for the first time by EASST. For this reason one of the PC-chairs (Hartmut Ehrig) and about 15 further members of EASST represented a strong European influence in the program committee of IDPT 2002. About 50% of the 10 invited lectures and 150 papers in 40 different sessions were presented by Europeans and contributed significantly to the high scientific standard of the conference. Moreover a best paper award was offered by EASST for the best software engineering paper:

### **Specifying Components for NEST applications**

by Asuman Sünül, Kestrel Institute, Berkeley, California

partly based on her PhD thesis at the TU Berlin. This paper has been part of the *Distinguished Lecture Series on Software Components for Continuous Engineering*.

Nearly 200 participants from all over the world have discussed their research and results, including topics as various aspects of process engineering and software engineering, including formal methods, architecture and components, design and simulation, and also applied topics as train control systems, health care, and medical applications.

Among the large number of high-quality and stimulating presentations there are the outstanding talks given by Professor M. Löwe (Invited Talk: *The Component Dilemma: Handicaps of Component Architectures in Commercial Information Systems*) and by Professor M. Arbib (Luncheon speaker: *Neural Computing, Round 3: Towards a Biologically Inspired Computer Architecture*). Both have in very dis-



immilar ways questioned the paradigm of components, that have been the topic in many sessions.

According to the tradition of this and several other American conferences there has been a large amount of parallelism concerning the accepted papers, as there always have been five sessions in parallel. Most participants - I guess - would have loved to split into two or three in order to attend all interesting sessions.

A fine and tasting conference dinner took place on Monday evening, where several awards were given to distinguished scientists that have specifically contributed to the high standards of this conference and the leadership within SDPS.

Cooperation between EASST and the Software Engineering Society (SES), a sub-society of of SDPS has been discussed and first steps for cooperation are about to be established.

Last but not least, let me point out to you the next IDPT conference:

### **IDPT 2003**

in Beijing, China, from June 16 - 20, 2003 (see <http://www.sdpsnet.org/>)



# 1st International Conference on Graph Transformation



**ICGT 2002 - Preliminary Program**  
Barcelona (Spain), October 7-12, 2002  
<http://www.lsi.upc.es/icgt2002>



## MONDAY October 7, 2002

09:00 - 18:30 *Satellite events: DNA & GRA - TRA // TERMGRAPH // GraBaTs*

## TUESDAY October 8, 2002

09:00 - 12:30 *Satellite events: GRA - TRA Tutorial // GraBaTs*

12:30 - 14:15 *Lunch Break*

14:15 - 14:30 *Opening of ICGT 2002*

14:30 - 15:30 **Graphical Calculi of Processes** (Invited Lecture)  
**Robin Milner**

15:30 - 16:00 *Coffee Break*

16:00 - 17:00 *Modelling and the use of tools (I)*

Transforming Specification Architectures by GenGED  
Roswitha Bardohl, Claudia Ermel and Julia Padberg

Computer Aided Multi-Paradigm Modelling to Process Petri Nets and Statecharts  
Juan de Lara and Hans Vangheluwe

17:30 - 19:00 *GRA - TRA Tool Session*



## WEDNESDAY October 9, 2002

**09:00 - 10:00** Ubiquitous, Decentralized, and Evolving Software:  
Challenges for Software Engineering (Invited Lecture)  
Carlo Ghezzi

**10:00 - 10:30** *Coffee Break*

**10:30 - 12:30** *Applications and Programming(I)*

Call-by-Value Lambda-Graph Rewriting — without Rewriting  
Maribel Fernández, Ian Mackie

Transformation: The Missing Link of MDA  
Anna Gerber, Michael Lawley, Kerry Raymond, Jim Steel and Andrew Wood

Abstraction and Control for Shapely Nested Graph Transformation  
Berthold Hoffmann

Graph-Based Reengineering of Telecommunication Systems  
André Marburger and Bernhard Westfechtel

**12:30 - 14:30** *Lunch Break*

**14:30 - 16:00** *Graph Algorithms and Graph Theory (I)*

Graph Transformations for the Vehicle Routing and Job Shop Scheduling Problems  
J. Christopher Beck, Patrick Prosser and Evgeny Selensky

Euler Graphs, Triangle-Free Graphs and Bipartite Graphs in Switching Classes  
Jurriaan Hage, Tero Harju and Emo Welzl

Decomposing Graphs with Symmetries  
Michel Bauderon and F. Carrere

**16:00 - 16:30** *Coffee Break*

**16:30 - 18:30** *Semantic and Theoretic Issues (I)*

Approximating the Behaviour of Graph Transformation Systems  
Paolo Baldan and Barbar König

Graph Transformation with Time: Causality and Logical Clocks  
Szilvia Gyapay, Reiko Heckel and Dániel Varró

Relabelling in Graph Transformation  
Annegret Habel and Detlef Plump

Confluence of Typed Attributed Graph Transformation Systems with Constraints  
Reiko Heckel, Jochen Malte Kuester and Gabriele Taentzer

**21:00 - 24:00** *Conference Dinner*



## THURSDAY October 10, 2002

**09:00 - 10:00** **Can Behavioural Requirements be Executed?**  
(and why would we want to do so? (Invited Lecture))

David Harel

**10:00 - 10:30** *Coffee Break*

**10:30 - 12:30** *Modelling and the use of tools (II)*

Unparsing of Diagrams with GiaGen

Mark Minas and Frank Strüber

Using Graph Transformation as the Semantical Model for Software Process Execution  
in the APSEE Environment

Carla A. Lima Reis, Rodrigo Quites Reis, Marcelo Abreu,

Heribert Schlebbe and Daltro J. Nunes

Gra-CAD – Graph-Based Tool for Conceptual Design

Janusz Szuba, Andy Schürr and Adam Borkowski

A Formal Semantics of UML Statecharts by Model Transition Systems

Dániel Varró

**12:30 - 14:30** *Lunch Break*

**14:30 - 16:00** *Applications and Programming (II)*

Formalising Behaviour Preserving Program Transformations

Tom Mens, Serge Demeyer and Dirk Janssens

Rule Invariants in Graph Transformation Systems for Analyzing Safety-Critical Systems

Julia Padberg and Bettina Enders

Incremental Transformation of Lattices: a Key to Effective Knowledge Discovery

John L. Pfaltz

**16:00 - 16:30** *Coffee Break*

**16:30 - 18:30** *Semantic and Theoretic Issues (II)*

Hyperedge Substitution in Basic Atom Replacement Languages

Renate Klempien-Hinrichs

Distributed Graph Transformation Units

Peter Knirsch and Sabine Kuske

Describing Policies with Graph Constraints and Rules

Manuel Koch and Francesco Parisi-Presicce

Linear Ordered Graph Grammars and Their Algebraic Foundations

Ugo Montanari and Leila Ribeiro



## **FRIDAY October 11, 2002**

**09:00 - 10:00** *Graph Algorithms and Graph Theory (II)*

Termination Detection of Distributed Algorithms by Graph Relabelling Systems

E. Godard, Y. Métivier, M. Mosbah and A. Sellami

Hierarchical Vertex Ordering

Sung-Ho Woo and Sung-Bong Yang

**10:00 - 10:30** *Coffee Break*

**10:30 - 12:30** *ICGT Panel and Closing*

**12:30 - 14:30** *Lunch Break*

**14:30 - 18:30** *Satellite events: GT - VMT // Software Evolution // Logic, GraTra & Discrete Structures*

## **SATURDAY October 12, 2002**

**09:00 - 12:30** *Satellite events: GT - VMT // Software Evolution // Logic, GraTra & Discrete Structures*

**12:30 - 14:30** *Lunch Break*

**14:30 - 18:30** *Satellite events: GT - VMT // Logic, GraTra & Discrete Structures*



**CALL FOR PARTICIPATION : ICGT 2002**  
**1st International Conference on Graph Transformation**  
**Barcelona (Spain), October 7-12, 2002**  
**<http://www.lsi.upc.es/icgt2002>**

The first international conference on graph transformation ICGT 2002 including several satellite events will be held in Barcelona in the second week of October 2002. It follows a series of six international workshops on graph transformation with applications in computer science held from 1978 to 1998 in Europe and the USA. The conference takes place under the auspices of EATCS, EASST, and IFIP WG 1.3. The proceedings of ICGT will be published by Springer-Verlag in Lecture Notes in Computer Science.

**Scope.** Graphical structures of various kinds (like graphs, diagrams, visual sentences and others) are very useful to describe complex structures and systems in a direct and intuitive way. These structures are often augmented by formalisms which add to the static description a further dimension allowing for the modelling of the evolution of systems via any kind of transformation of such graphical structures. The field of Graph Transformation is concerned with the theory, applications and implementation issues of all these formalisms.

The theory is strongly related to areas such as graph theory and graph algorithms, formal language and parsing theory, theory of concurrency and distributed systems, formal specification and verification, logic and semantics. The application areas include all those fields of Computer Science, Information Processing, Engineering and Natural Sciences where static and dynamic modeling by graphical structures and graph transformations, respectively, play an important role. In many of these areas tools based on graph transformation technology have been implemented and used.

**Invited speakers** Carlo Ghezzi (Milano, Italy) David Harel (Rehovot, Israel) Robin Milner (Cambridge, UK)

**Program committee.** Michel Bauderon (Bordeaux, France), Paolo Bottoni (Rome, Italy), Andrea Corradini (co-chair; Pisa, Italy), Hartmut Ehrig (Berlin, Germany), Gregor Engels (Paderborn, Germany), Reiko Heckel (Paderborn, Germany), Dirk Janssens (Antwerp, Belgium), Hans-Jörg Kreowski (co-chair; Bremen, Germany), Ugo Montanari (Pisa, Italy), Manfred Nagl (Aachen, Germany), Fernando Orejas



(Barcelona, Spain), Francesco Parisi-Presicce (Rome, Italy), Mauro Pezzé (Milano, Italy), John Pfaltz (Charlottesville, Virginia, USA), Rinus Plasmeijer (Nijmegen, The Netherlands), Detlef Plump (York, Great Britain), Azriel Rosenfeld (Maryland, USA), Grzegorz Rozenberg (Leiden, The Netherlands), Andy Schürr (Munich, Germany), Gabriele Taentzer (Berlin, Germany), Gabriel Valiente (Barcelona, Spain)

**General Organizing Committee.** Andrea Corradini (Pisa, Italy), Hartmut Ehrig (chair; Berlin, Germany), Hans-Jörg Kreowski (Bremen, Germany), Fernando Orejas (Barcelona, Spain), Grzegorz Rozenberg (Leiden, The Netherlands)

**Local Organizing Committee.** Nikos Mylonakis, Fernando Orejas (chair), Elvira Pino, Gabriel Valiente

**Steering committee.** Michel Bauderon (Bordeaux, France), Andrea Corradini (Pisa, Italy), Hartmut Ehrig (chair; Berlin, Germany), Gregor Engels (Paderborn, Germany), Dirk Janssens (Antwerp, Belgium), Hans-Jörg Kreowski (Bremen, Germany), Ugo Montanari (Pisa, Italy), Manfred Nagl (Aachen, Germany), Francesco Parisi-Presicce (Rome, Italy), John Pfaltz (Charlottesville, Virginia, USA), Rinus Plasmeijer (Nijmegen, The Netherlands), Azriel Rosenfeld (Maryland, USA), Grzegorz Rozenberg (Leiden, The Netherlands)

More details concerning ICGT 2002 including registration to the main conference and satellite events can be found on the website of ICGT 2002,

<http://www.lsi.upc.es/icgt2002>

For further information, you may contact also [icgt2002@lsi.upc.es](mailto:icgt2002@lsi.upc.es).

**Conference address.** ICGT 2002 Fernando Orejas Universitat Politècnica de Catalunya Departament de Llenguatges i Sistemes Informàtics Campus Nord - Edif. C6 08034 Barcelona, Spain Tel: +93 401 7018 Fax: +93 401 7014

#### **Satellite events.**

**GRA-TRA TUTORIAL** Tutorial on Foundations and Applications of Graph Transformation Date: Oct. 8 (morning) Organizers, contact and further information: Luciano Baresi ([baresi@elet.polimi.it](mailto:baresi@elet.polimi.it)), Reiko Heckel ([reiko@upb.de](mailto:reiko@upb.de)) <http://www.upb.de/cs/ag-engels/Conferences/ICGT02/Tutorial>

**DNA GRA-TRA 2002** Tutorial on DNA Computing and Graph Transformation Date: Oct. 7 Organizers: Tero Harju (Turku, Finland), Grzegorz Rozenberg (Leiden, The Netherlands), Ion Petre (Turku, Finland). Contact and further information: [rozenber@liacs.nl](mailto:rozenber@liacs.nl)

**TERMGRAPH 2002** International Workshop on Term Graph Rewriting Date: Oct. 7 Organizer, contact and further information: Detlef Plump ([det@cs.york.ac.uk](mailto:det@cs.york.ac.uk)) [http://www.cs.york.ac.uk/~det/Termgraph\\_2002/cfp.html](http://www.cs.york.ac.uk/~det/Termgraph_2002/cfp.html)



**GraBaTs 2002** International Workshop on Graph-Based Tools Date: Oct. 7 - 8 (lunch) Organizers: Tom Mens (Brussels, Belgium), Andy Schürr (Munich, Germany), Gabriele Taentzer (Berlin, Germany) Contact and further information: [gabi@cs.tu-berlin.de](mailto:gabi@cs.tu-berlin.de), <http://tfs.cs.tu-berlin.de/grabats> .

**GT-VMT 2002** International Workshop on Graph Transformation and Visual Modeling Techniques Date: Oct. 11 (lunch) - Oct. 12 Organizers: Paolo Bottoni (Rome, Italy), Mark Minas (Erlangen, Germany) Contact and further information: [bottoni@dsi.uniroma1.it](mailto:bottoni@dsi.uniroma1.it), [mark.minas@informatik.uni-erlangen.de](mailto:mark.minas@informatik.uni-erlangen.de) <http://www2.cs.fau.de/GTVMT02/> .

**SOFTWARE EVOLUTION** Workshop on software evolution through transformations: Towards uniform support throughout the software life-cycle Date: Oct. 11 (lunch) - Oct. 12 (lunch) Organizers: Reiko Heckel (Paderborn, Germany), Tom Mens (Brussels, Belgium), Michel Wermelinger (Lisbon, Portugal) Contact and further information: [reiko@upb.de](mailto:reiko@upb.de), <http://www.upb.de/cs/ag-engels/Conferences/ICGT02/FSWE>

**LOGIC, GRAPH TRANSFORMATIONS AND DISCRETE STRUCTURES** Workshop with invited lectures and short contributions Date: Oct. 11 (lunch) - Oct. 12 Organizers: Bruno Courcelle (Bordeaux, France), Pascal Weil (Bordeaux, France) Contact and further information: [pascal.weil@labri.fr](mailto:pascal.weil@labri.fr), <http://dept-info.labri.fr/~weil/LGTDD-Barcelona2002> .



---

## **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