Testing of Event-Triggered Real-Time Systems

May 17, 1999

Summary

Applying institution: Institutionen för Datavetenskap, Högskolan i Skövde.
Co-applicants: Sten F. Andler, Ragnar Birgisson, Jonas Mellin.
Partner Company: ENEA Data AB

The current industry practice for testing real-time applications is often based on case specific ad-hoc techniques. At the same time real-time systems are often employed in environments where their correctness is of great importance, which necessitates a rigorous approach to testing. In this context, the ability to test all anticipated behaviors is strongly desirable, but in general not feasible. Our approach is based on the hypothesis that the testability of event-triggered real-time systems can be improved by applying constraints on application behavior. This makes it possible to define an upper bound on the test effort, which enables full testing. However, in a system that uses constraints to bound the test effort, it is often not possible to perform all the necessary testing within the allotted time frame. Therefore, test case selection methods must be employed. Moreover, the criteria for such methods must be specially suited for real-time systems, and measurement techniques for evaluating the quality of the selected test cases must be produced. Currently, the industry uses manual or semi-automated test case execution techniques. In this project automated tools for test case generation, selection, and execution are produced.

Three Licentiate/Ph.D. students from Skövde will be actively working on the project[1]. In addition members of the ENEA Test group will contribute to this project through their experience and insight into current industry practice in testing of real-time systems. Further, ENEA Data's broad range of industry contacts provides an excellent opportunity to transfer the technology produced by this project into the industry.

1 Problem Statement

Software testing is a fundamental aspect of software development, and can consume as much as 50% of the total cost for a project. This is particularly true for real-time systems, which have to be rigorously tested to ensure their correctness. Further, real-time systems must not only be tested for functional correctness, but also for timeliness.

To achieve the desired level of confidence in application correctness, full test coverage is strongly desirable for real-time systems. This means that all anticipated behaviors must be tested. In general, full test coverage is not feasible, because of the large amount of test cases that must be designed, executed, and analyzed. Therefore, to enable testing with full test coverage, some constraints on application behavior must be introduced. One example of this are time-triggered real-time systems [Kop91]. However, there are situations where the time-triggered approach is not suitable, the dynamic nature of some systems necessitates the use of an event-triggered architecture.

This project investigates an alternative approach that aims at bounding the test effort for event-triggered systems, and thus, enabling full testing of such systems. The basis of this approach is to

[1] We regret not to have meet the 5 page limit for this application, but since it describes three Ph.D. projects we found it difficult address them in the allotted space.

use a system architecture that inherits the testability of time-triggered systems while still allowing event-triggered semantics. Our approach focuses on system level testing of timeliness, assuming that functional correctness has already been tested. This can reveal incorrect assumptions about worst-case execution times and errors in schedulability analysis.

To improve the testability of event-triggered systems, three major problem areas must be addressed. The first problem area is related to identifying constraints on application behavior that reduce the required test effort. The effects of these constraints on the application and the execution environment must be investigated. The second problem area is related to generation and selection of test cases. Based on the constraints placed on application behavior, it must be possible to generate all the required test cases. Further, since the number of test cases may be very high, this process must be automated, and methods for test case selection must be produced. Finally, the third problem area concerns test case execution. A test case execution strategy must be developed, and automated tools for executing test cases and analyzing the results must be provided.

2 Main Ideas

2.1 Theoretical Foundation

Our approach is based on the hypothesis that the testability of event-triggered real-time systems can be improved by applying constraints on application behavior. Recent work by Mellin [Mel99] has suggested three constraints; (i) a bound on the number of times a task can be preempted; (ii) a bound on the number of concurrently executing tasks; and (iii) constraining the points in time where the environment can be observed. Using these constraints, an upper bound on the test effort for event-triggered real-time systems is defined by Mellin, and refined by Birgisson [Bir98]. Further, an initial study of the effects that these constraints have on scheduling, and how they can be enforced has been performed [Bir98].

The aim of this project is to further strengthen the theoretical foundation for the hypothesis stated above. To accomplish this, the following objectives have been identified: (i) investigation of the constraints and assumptions made by Mellin and Birgisson; (ii) formulation of an improved upper bound on the test effort based on the selected constraints and assumptions; (iii) refinement of the current enforcement methods; and (iv) identification and solution to problems that arise in a constrained application.

2.2 Test Case Generation and Selection

Given an upper bound on the required test effort for event-triggered real-time systems, it is possible to generate all the test cases required for full test coverage. This requires defining the contents of a test case, and initial work on this has been done by Birgisson [Bir98] and Nilsson [Nil99]. The basic idea is to define a test case as consisting of the state of the system (represented by the states of active tasks) and possible future events. In addition, event parameters and the state of shared resources must be included. The combination of each system state and future events represents all anticipated behaviors of the system, and it is possible to determine if tasks miss their deadlines as a result of any of these behaviors.

While the proposed approach bounds the number of test cases, the upper bound may still be very high, and thus, it is not feasible to execute all test cases. Therefore, test case selection must be used to select a subset of the test cases. In the context of our approach, test case selection is based on identifying worst-case scenarios, i.e., system states that are likely to result in failure to meet timing constraints. Methods for measuring the quality of the selection test cases must also be produced.

The aim of this project is to identify techniques that can, according to a specific criteria, automatically create all the required test cases. To achieve this, the following objectives have been identified: (i) refinement the contents of a test case; (ii) development an algorithm that creates all

the test cases based on the upper bound test effort; (iii) identification of test selection criteria and methods for applying these; (iv) specification, design, and implementation of automated tools for generation and selection of test cases.

2.3 Test Case Execution

In our approach test case execution uses the information in a test case description to put the system into the specified state, and presents the specified events to the system. Using a test oracle the expected results are compared to the produced output. The results of a test execution are monitored and recored for future reference.

The aim of this project is to automate the test case execution process in a system using the suggested constraints for improved testability. This requires achieving the following objectives: (i) refinement of Nilsson's test case execution strategy; (ii) specification, design, and implementation of a test driver that uses this test case execution strategy; (iii) specification, design, and implementation of constraints enforcement mechanisms; (iv) performing a case study of the proposed approach using an application running on the DeeDS prototype [And96].

3 Expected Results and Impact

3.1 Theoretical Foundation

The expected contributions of this project are:

1. A solid theoretical foundation for using constraints on application behavior for improving testability.
2. Methods for enforcing the constraints and solutions to problems that arise in a constrained application.
3. Integration of problem specific solutions and enforcement methods into the DeeDS prototype.

A solid theoretical foundation, for using constraints on application behavior to improve testability, is essential for establishing a common understanding of the implications, effects, and potential advantages of such an approach. In a constrained system, mechanisms for enforcing the constraints must be in place. A constrained system without these enforcement mechanisms cannot take full advantage of our approach, since no guarantees can be given with respect to test coverage. It is important to note, however, that even without the guarantees given by the enforcement mechanisms our approach is still superior to current industry practice.

3.2 Test Case Generation and Selection

The expected contributions of this project are:

1. Identification of test case selection criteria and methods for applying these criteria.
2. Methods for measuring the quality of the selected test cases.
3. A test case generation and selection strategy for an application using the suggested constraints that is based on the above criteria.
4. Automated tools for generation and selection of test cases based on the suggested strategy.

Given an upper bound on the test effort for event-triggered real-time systems, the results of this project will improve current methods for test case selection. Our approach makes it possible to determine the proportion of system behaviors that have been tested, and provides test case selection methods that are specifically tailored for real-time systems. Further, the ability to measure the quality of the selected test cases makes is easier to determine when a system has been sufficiently tested. Currently, the degree to which a system is tested is determined by the

resources allocated to the testing process, and the current test coverage metrics do not consider the quality of the selected test cases.

3.3 Test Case Execution

The expected contributions of this project are:

1. A complete test case execution strategy for a system using the suggested constraints.
2. Automated tools for executing test cases and analyzing the results of test runs.
3. A case study that evaluates the suggested approach.

Currently, the industry relies and manual or semi-automatic techniques for test case execution. This project will provide automated tools, that execute a large series of test cases, compare the expected results with the actual output, and monitor how the results were reached. This will increase the number of test cases that can be executed in the allotted time, and hence, support more rigorous testing of real-time applications. Further, manual test case execution is error prone (due to the need for human intervention), which is not the case with automated techniques.

4 Project Plan

This section presents a project plan for each of the sub-projects, consisting of a short description followed by a more detailed plan in table 4.1. Dates in boldface represent deliverables. The duration of these projects is 5 years (4 years effective time).

Theoretical Foundation

The first phase of the project will be concerned with analysis of current methods for improving testability by using constraints on application behavior. The second phase focuses on refining these methods, and specifying enforcement mechanisms for such a system. Solutions to specific problems that arise in a constrained application, and an enforcement mechanisms will be implemented and integrated into the DeeDS prototype.

Test Case Generation and Selection

The first phase of this project will be concerned with analysis of current test case selection techniques. It will also identify suitable criteria for test case selection in a real-time system, and methods for measuring the quality of the selected test cases. The second phase focuses on producing a test case selection strategy for a system using the proposed constraints on application behavior. The test case selection strategy will provide the foundation for specifying, designing, and implementing a prototype of an automated test case generation and selection tool.

Test Case Execution

The first phase of this project will be concerned with analysis of current test case execution techniques for real-time systems. The second phase focuses on producing a test case execution strategy for a system using the proposed constraints on application behavior. The execution strategy will provide the foundation for specifying, designing, and implementing a prototype to a test case execution tool. An existing pilot application will then be used to perform a case study of our approach on the DeeDS prototype.

	Theoretical Foundation	Test Case Selection	Test Case Execution
1999-10 — 1999-12	Ph.D. Student recruitment	Ph.D. Student recruitment.	Ph.D. Student recruitment.
1999-12 — 2000-12	Analysis of current methods for improving testability.	Analysis of current methods for test case generation and selection.	Analysis of current test case execution methods.
2000-12	Report on methods for improving testability.	State-of-the-art report on test case generation and selection methods.	State-of-the-art report on test case execution methods.
2000-12 — 2001-12	Development of refined upper bound. Specification of enforcement mechanisms.	Development of test case generation and selection strategy.	Development of test case execution strategy.
2001-12	Intermediate report and updated project specification.	Intermediate report and updated specification of the project.	Intermediate report and updated specification of the project.
2001-12 — 2003-12	Implementation of enforcement mechanism. Solutions to application specific problems.	Development of test case generation and selection tools.	Development of test case execution tool. Case study.
2003-12	Enforcement mechanisms and solutions to application specific problems integrated into DeeDS.	Automated tools for generating, selecting, and analyzing the quality of test cases.	Tools for test case execution and analysis. Results of case study
2003-12 — 2004-10	Final write-up of Ph.D. thesis	Final write-up of Ph.D. thesis.	Final write-up of Ph.D.thesis.
2004-10	Ph.D. thesis.	Ph.D. thesis.	Ph.D. thesis.

Table 4.1. Project Plans

5 Preliminary Budget

We request for funding of three PhD students. These students will allocate 80% of their time to this project, and thus, are expected to complete their Licentiate/Ph.D degree in 2.5 and 5 years respectively. The estimated costs in SEK are shown is the table 5.1.

	Cost over 5 years					
YEAR		Year 1	Year 2	Year 3	Year 4	Year 5
Salary cost, Ph.D. students	4760600	894600	894600	926600	971300	1073500
Salary cost, project leader etc.	2171000	434200	434200	434200	434200	434200
Travel	336000	36000	66000	66000	84000	84000
Administrative overhead	1595340	299000	306180	313200	326960	349400
TOTAL COST	8862940	1664400	1700980	1740000	1816460	1941100

Table 5.1. Estimated project budget (SEK).

6 Related Research

MARS is a framework for time-triggered distributed real-time systems, developed at the Technical University of Vienna [Kop89]. A complete test strategy and the necessary tools have been produced for the MARS approach [Sch90]. This approach uses the inherent constraints of time-triggered systems to improve testability. In contrast, our approach aims at improving the testability of event-triggered systems, and focuses on testing real-time applications for timeliness.

Methods for reducing the test effort through test case selection range from general approaches such as category partitioning [ABC82, GG75, OB87], to more specific ones based on finite state machines [FB+91], and statistical methods that randomly select inputs from an input space [DN84]. In contrast, our approach is based on reducing the test effort, while still maintaining full test coverage.

A number of methods that deal with test case execution have appeared in recent years, e.g., Peleska and Siegel [PS97] and Cardell-Oliver and Glover [CG98]. However, none of these are based on the kind of state information our approach suggest. Moreover, these methods do not explicitly address full test coverage, while our approach can guarantee that all expected behaviors have been tested.

7 Relation to the Profile

This project is related to three aspects of the profile, dependability, distributed systems, and real-time database systems. Our approach will enable full testing of real-time applications which can make them more dependable. In situations where full testing is not feasible, our approach will still support greater test coverage than possible in an unconstrained system. Further, the use of constraints on application behavior, and our methods for measuring the quality of selected test cases, provide a better coverage measurement technique than currently available.

Our approach is primarily aimed at distributed active real-time database systems, since they provide many of the facilities required for implementing the test tools. The approach is, however, not restricted to such an architecture, and can be applied to any system where similar services are made available.

8 Industrial Relevance

One of the main focuses of the ENEA Test department is to actively contribute to the test methods and processes used by their customers in various industries. Current industrial practice in testing real-time systems is often based on case specific ad-hoc techniques. There is a distinct lack of a solid methodology for testing real-time systems. Therefore, it is of great importance to ENEA Test to be able to provide its customers with test methods and algorithms which both improve the quality of testing and at the same time facilitate the measuring of the test quality. The customers of ENEA Test cover a large proportion of the Swedish real-time industry, which provides an excellent opportunity for transferring the technology advances of this project out to the industry.

9 Context

This research will be carried out in the Distributed Real-Time Systems (DRTS) research group at the University of Skövde which is lead by Professor Sten F. Andler, Department of Computer Science, University of Skövde, BOX 408, S-541 28, Skövde, tel. 0500-464713, email sten@ida.his.se. The group consists of two senior researchers, and four PhD candidates. The DRTS group is currently developing a prototype for an active real-time database system (DeeDS) in a NUTEK funded project (since 1994) [And96]. From the start this project has been conducted in cooperation with ENEA Data and ENEA OSE Systems.

ENEA Data has been involved with real-time systems for more than 30 years, both as a product development company and a consultant company. ENEA Data has around 450 employees and its turnover was 380 MSEK in 1998. ENEA's main product is the real-time operating system (RTOS) OSE, which is the market leading RTOS in Scandinavia and one of the leading RTOS in other European countries. ENEA Test has around 50 consultants working with the industry, testing both hardware and software. ENEA Test is lead by Thomas Vesterlund, Affärsenhetschef, BOX 232 ,18323 Täby, tel. 08-50714000, fax 08-50714040, email thve@enea.se.

ENEA Data's contribution to this project is twofold. Firstly, their consultants possess extensive knowledge on industry practice in testing real-time systems, which provides valuable input to this project. Secondly, their extensive range of contacts within the industry provides a platform for validating the methods produced in this project by testing them with various clients.

A support letter from ENEA Data is not included in this application, but it will be sent as soon as possible.

References

[ABC82] Adrion, W.R., Branstad, M.A., and Cherniavsky, J.C. Validation, Verification, and Testing Computer Software. ACM Computing Surveys 14, June 1982, pp. 159-192.

[And96] Andler S.F., et. al. DeeDS Towards a Distributed and Active Real-Time Database System. ACM SIGMOD Record, Special Issue on Real-Time Database Systems, 1996.

[Bir98] Birgisson, R. Improving Testability of Applications in Active Real-Time Database Environments. Masters thesis, Department of Computer Science, University of Skövde, 1998.

[BMA99] Birgisson R., Mellin J., and Andler S.F. Bounds on Test Effort for Event-Triggered Systems. Technical Report HS-IDA-TR-99-001, University of Skövde, Department of Computer Science, 1999.

[CG98] Cardell-Oliver, R., and Glover, T. A practical and Complete Algorithm for Testing Real-Time Systems. Technical Report CSM-306, Department of Computer Science, University of Essex, 1998.

[DN84] Duran, J.W., and Naftos, S.C. An Evaluation of Random Testing. IEEE Transactions on Software Engineering 10(4), July 1984.

[GG75] Goodenough, J.B., and Gerhart, S.L. Towards a Theory of Test Data Selection. IEEE Transactions on Software Engineering SE-2, June 1975, pp 156-173.

[FB+91] Fujiwara, S.F., Bochman, G.V., Khendek, F., Amalou, M., Ghedamsi, A. Test Selection Based on Finite State Models. IEEE Transactions on Software Engineering 17(6), June 1991.

[Kop91] Kopetz, H. Even-Triggered Versus Time-Triggered Real-Time Systems. Lecture Notes in Computer Science, Vol 563, pp 87-101. Springer-Verlag, Berlin/New York, 1991.

[Kop89] Kopetz et. al. Distributed Fault-Tolerant Real-Time Systems: The MARS Approach. IEEE Micro 9, pp. 25-40, Feb. 1989.

[Lin99] Birgitta L. Reducering av Testfall vid Testning av Punktlighet i Realtidssystem. Final Year Project, University of Sköve, 1999.

[Mel99] Mellin, J. Supporting System Level Testing Of Applications by Active Real-Time Database Systems. Proceedings of the Second International Workshop on Active, Real-Time, and Temporal Database Systems. LNCS 1553, Springer-Verlag. 1999.

[Nil99] Nilsson, R. Automated Test Case Execution for Real-Time Systems Constrained for Improved Testability. Final Year Project, University of Skövde, 1999.

[OB87] Ostrand, T.J., and Balcer, M.J. The Category-Partitioning Methods for Specifiying and Generating Functional Tests. Submitted for "Special Section on Software Testing" to Communications of the ACM.

[PS97] Peleska J., and Siegel, M. Test Automation of Safety-Critical Reactive Systems. South African Computer Journal 19, pp. 53-73, 1997.

[Sch90] Schutz, W. A Test Strategy For the Distributed Real-Time System MARS. Proceedings of IEEE International Conference on Computer Systems and Software Engineering. New York IEEE Coorperation, pp 20-27, 1990.

Appendix A - CV's for Applicants

A.1. Curriculum Vitae for Sten F. Andler

Professor of Computer Science
Dept. of Computer Science, Högskolan i Skövde
PO Box 408, S-541 28 Skövde, Sweden

Research Experience

Högskolan i Skövde (University of Skövde), Dept. of Computer Science, Skövde, Sweden.
Apr. 1993 - present, Professor of Computer Science (bitr. prof. i datalogi)

Distributed Real-Time Systems Research Group: This research group was formed in September 1993 as a competence center for database-related problems in distributed real-time systems, such as synchronization, scheduling, reactive mechanisms, debugging, distribution, etc. We are actively carrying out studies and research in these areas, and are involved in the externally funded industrial research collaboration projects below, which have led to a Distributed Active Real-Time Database Systems prototype and research vehicle called DeeDS.

Distributed Reconfigurable Real-Time Database Systems Project: This is a collaboration with ENEA Data AB, Täby, Sweden, and is sponsored by NUTEK, the National Board for Industrial and Technical Development, in February 1994. Our short-term goal is to build an environment in which we can experiment freely with the components of a well-designed, flexible, distributed real-time database system. Our long-term goal is to use this experimental system as a base on which to pursue research in dynamic real-time scheduling, reactive real-time databases, and distributed real-time debugging.

IBM Software Solutions Division, Santa Teresa Laboratory, San Jose, California
Feb. 1992 - Apr. 1993, Advisory Programmer.

DataHub/AIX Future Project: Defined and specified a future product to allow remote management of relational databases on AIX/6000 (a UNIX operating system running on RISC System/6000) from a DataHub workstation anywhere in the network, in a similar way that was envisioned for control of databases on the mainframe.

Information Warehouse Design Control: Participated in the Information Warehouse Architecture group, and co-produced an internal Information Warehouse Design Control document, proposing the directions for a set of early products in the Information Warehouse series, whose intention it was to make all of an enterprise's data available in understandable form to knowledge workers anywhere in the organization.

IBM Research Division, Almaden Research Center, San Jose, California
Oct. 1979 - Feb. 1992, Research Staff Member.

Electronic Imaging/Font Technology: Developed imaging engines for high-function printers and displays. Invented techniques for automatic derivation of gridfitting hints for scaleable fonts. Implemented an outline font rasterizer for Adobe Type 1 font format. This involved design and implementation of techniques for processing font hints and continuity requirements, in order to render professional-quality scaleable fonts. This Type I font rasterizer was donated by IBM, in

Oct. 1991, to the MIT X Consortium for distribution as part of the X Windows System, the industry standard windowing environment for UNIX workstations. Nov 1985 - Feb 1992.

Technical Staff: Reviewed budgets and research plans for the Computer Science Department. Conducted technical reviews of Computer Science Projects. Participated in Corporate Task Force to determine what the Research Division could do to support the company's printer and display divisions. Nov. 1984 - Nov. 1985.

Workstation System Software: Developed systems software for a family of office workstations. Designed and implemented communication software. Operated electronic mail gateway to CSNET/Internet and redesigned electronic mail software. Sep. 1993 - Nov. 1984.

System D: Developed a distributed transaction processing system, consisting of a set of servers connected by a high-speed local network. Designed the system architecture and created the communication subsystem. Oct. 1979 - Jun. 1981.

Xerox Palo Alto Research Center (PARC), Palo Alto, California.
May 1977 - Aug. 1977, Research Intern.

Modularized an interactive graphics editor, and then added a communications module to allow collaboration between any number of users on an Internet (containing both Ethernets and slow telephone lines). The design was a simple distributed database, where the requirements were to maintain consistent copies of a small fast-changing data base at nodes of a minimally connected network of virtual circuits. Updates could occur at any node, and nodes or groups of nodes could leave and enter the collaboration network.

Carnegie Mellon University, Dept. of Comp. Sci., Pittsburgh, Pennsylvania.
Aug. 1975 - Aug. 1979, Graduate Assistant.

Research on synchronization mechanisms, in particular, the specification, implernentation, and verification properties of Predicate Path Expressions. Designed and implemented Predicate Path Expressions and a data abstraction mechanism as an Algol 68 preprocessor. Designed and implemented an experimental programming language using the CDL-2 compiler-compiler.

Chalmers University of Technology, Göteborg, Sweden.
Feb. 1972 - Aug. 1975, Teaching/Research Assistant.

Analyzed and simulated cyclic storage media, especially the interference between input-output processes and the effect of buffering.

Designed microprogramming support to implement a control structure for small computers.

Degrees

Ph.D. in Computer Science, Chalmers University of Technology, Göteborg, Sweden. Dec. 1979.

Ph.D. in Computer Science, Carnegie Mellon University, Pittsburgh, Pennsylvania. Aug. 1979.

M.S. in Computer Science, Carnegie Mellon University, Pittsburgh, Pennsylvania. May 1976.

M.S. in Engineering, Chalmers University of Technology, Göteborg, Sweden. Feb. 1972.

Teaching Experience

Högskolan i Skövde (University of Skövde), Dept. of Computer Science, Skövde, Sweden.
Apr. 1993 - present, Professor of Computer Science (bitr. prof. i datalogi)

IBM Research Division, Almaden Research Center, San Jose, California.

University of California at Berkeley, Berkeley, California.
Spring 1984, Visiting Professor of Computer Science.

Carnegie Mellon University, Dept. of Comp. Sci., Pittsburgh, Pennsylvania.
1975 - 1979, Graduate Assistant.

Chalmers University of Technology, Göteborg, Sweden.
Fall 1976, Teaching assistant.
1972 - 1975, Instructor and Teaching Assistant.

Honors and Awards

Research Division Technical Group Award, IBM Almaden Research Center, Dec. 1993. For "contributions to the development of the Hourglass language and image processing software for use with printers and displays."

External Recognition Award (Secretary/Treasurer to ACM SIGOPS), IBM Almaden Research Center, Nov. 1988.

Research Informal Award (Hosted First Corporate Datastream Architecture Meeting), IBM Almaden Research Center, Jan 1987.

Research Division Award (Developed electronic mail software and a gateway to CSNET/Internet), IBM Research Division, May 1983.

Graduate Stipend, Carnegie Mellon University, Aug. 1975 - Aug. 1979.

Doctoral Scholarship, Chalmers University of Technology, 1975 - 1979.

Travel Fellowship (for studies in the USA), The Swedish Government, 1975.

ASEA Scholarship (for studies in the USA), The Sweden-America Foundation, 1975.

Professional Activities

Member of the Editorial Board of the IEEE Press Series on Engineering of Complex Computer Systems

Member of the Executive Committee of the IEEE Technical Committee on Complexity in Computing (TCCX)

Program Co-Chair of the Fourth International Conference on Engineering of Complex Systems (ICECCS '98), Monterey, California.

Publicity co-chair of the Real-Time Technology and Applications Symposium (RTAS '98).

Member of the Program Committee of the Third International Conference on Engineering of Complex Systems (ICECCS '97), Como, Italy.

Co-Chair of the Complex Database Systems and Data Management Track of the Second International Conference on Engineering of Complex Systems (ICECCS '96), Montreal, Canada.

Member of the Program Committee of the Second International Conference on Engineering of Complex Systems (ICECCS '97), Montreal, Canada.

Member of the Program Committee of Real-Time Database Workshop RTDB '96, USA.

Program Co-Chair of the International Workshop on Active and Real-Time Database Systems (ARTDB-95), Skövde, June 1995. The workshop was co-sponsored by ACM and ACT-NET, the European Research Network for Active Databases.

Member of Program Committee, International Workshop on Raster Imaging and Digital Typography (RIDT'89), Lausanne, Switzerland, Oct. 12-13, 1989.

Secretary/Treasurer to ACM SIGOPS for two years.

Memberships in Professional Organizations

Association for Computing Machinery (ACM)

IEEE Computer Society

Publications

1. "DeeDS Towards a Distributed Active and Real-Time Database System," with Jörgen Hansson, Joakim Eriksson, Jonas Mellin, Mikael Berndtsson and Bengt Eftring. ACM SIGMOD Record, Special Section on Advances in Real-Time Database Systems, 25, 1 (March 1996), pp. 38-40.

2. "Refining Design Constraints from a System Services Model of a Real-Time DBMS," with J. Mellin, J. Hansson. In Proc. 1st Int'l Workshop on Real-Time Databases: Issues and Applications, March 1996.

3. "Automatic Generation of Gridfitting Hints for Rasterization of Outline Fonts and Graphics." In EP90: Proc. Int'l Conf. Electronic Publishing, Document Manipulation & Typography, R. Furuta (ed.), Cambridge University Press, 1990, pp. 221-234.

4. "The COSIE Communication Subsystem: Support for Distributed 0ffice Applications," with D. B. Terry. ACM Trans. Office Information Systems (TOOIS) 2, 2(Apr. 1984), pp. 79-95.

5. "Experience with Measuring Performance of Local Network Communications," with D. B. Terry. In Proc. 26th IEEE Computer Society Int'l Conf. (COMPCON), Feb. - Mar. 1981, pp. 203-205.

6. "System D: A Distributed System for Availability," with I. Ding, K. Eswaran, C. Hauser, W. Kim, J. Mehl, and R. Williams. In Proc. 8th Int'l Conf. Very Large Data Bases (VLDB), Mexico City, Sep. 1982, pp. 33-44.

7. "On Enhancing Local Network Communication Devices," with D. Daniels and A. Spector. In Local Networks for Computer Communications, A. West and P. Janson (eds.). North-Holland Publishing Company, 1981, pp. 191-205.

8. "Predicate Path Expressions: A High-Level Synchronization Mechanism." Ph.D. Thesis, Carnegie Mellon University, Pittsburgh, Pennsylvania, Aug. 1979.

9. "Predicate Path Expressions." In Conf. Record 6th ACM Symposium on Principles of Programming Languages (POPL), San Antonio, Texas, Jan. 1979, pp. 226-236.

10. "Synchronization Primitives and the Verification of Concurrent Programs." In Operating Systems: Theory and Practice, D. Lanciaux (ed.). North-Holland Publishing Company, 1979, pp. 67-99.

11. "Types in Algol 68," with Peter Hibbard. In Proc. 5th Annual III Conf. on the Implementation and Design of Algorithmic Languages, IRIA, Guidel, France, May 1977, pp. 124-144.

Technical Reports

1. "DeeDS: A Distributed Active Real-Time Database System," with M. Berndtsson, B. Eftring, J. Eriksson, J. Hansson and J. Mellin. Technical Report HS-IDA-TR-95-008, Högskolan i Skövde, Skövde, Sweden, June 1995.

2. "The Distributed Reconfigurable Real-Time Database Systems Project," with J. Hansson, J. Eriksson, and J. Mellin. Technical Report HS-IDA-TR-94-006, Högskolan i Skövde, Skövde, Sweden, Sep. 1994.

3. "925 Workstation, Level 1 System: Communication Subsystem (CSS) and Simple Name Service (SNS)," with D. B. Terry. Internal Report, IBM Research Laboratory, San Jose, California, Jan. 1983.

4. "Research Computing Environment," with R. P. O'Hara, C.W. Christensen, M.R. Korn, and D.A.H. Smith. Task Force Report, IBM Research Laboratory, Yorktown, New York, Nov. 1981.

5. "Letter to the Editor, with Peter Feiler, A. N. Habermann, V.R. Prasad, and Walter Tichy. Operating Systems Review 12, 1 (Jan 1978), 6-11.

6. "Studies and Experiences in the USA 1975-76." Technical Report 76.06, Dept. of Computer Sciences, Chalmers University of Technology, Göteborg, Sweden, Nov. 1976.

7. "A Software Oriented Implementation of a Control Structure for Small Computers," with Hans Lindström, Leif Svensson, and Iris Ödman. Internal Report, Department of Computer Sciences, Chalmers University of Technology, Göteborg, Sweden, Jun. 1975

8. "A Program for Elimination of Zero Elements in the Diagonal of a Matrix by Permutation of Rows." Technical Report 74.04, Department of Computer Sciences, Chalmers University of Technology, Göteborg, Sweden, 1974.

9. "Automatic generation of IBM/360 Job Control Language." Technical Report 72.02, Department of Computer Sciences, Chalmers University of Technology, Göteborg, Sweden, 1972.

A.2. Short Curriculum Vitae for Jonas Mellin

Department of Computer Science, Högskolan i Skövde
Box 408, S-541 28 Skövde, Sweden

Degrees

Lic. Phil. Predictable Event Monitoring, University of Linköping, 1998.
M.Sc. of Computer Science in Artificial Intelligence and Software Engineering, Department of Computer Science, Exeter University, United Kingdom, 1989.
Member of the Distributed Real-Time Systems research group since 1993. Formally signed up as a Ph.D. student at University of Linköping.

Research Interests

Research interests are mainly multi-purpose timely event monitoring and its uses. Further, interests are in software engineering, requirements engineering (elicitation, negotiation, and validation), distributed systems, databases, object-orientation, software architecture, testing, debugging, and programming methodology.

A.3. Short Curriculum Vitae for Ragnar Birgisson

Department of Computer Science, Högskolan i Skövde

Box 408, S-541 28 Skövde, Sweden

Degrees

M.Sc. of Computer Science in New Generation Representations, Department of Computer Science, University of Skövde, Sweden, 1998.
B.Sc. in Computer Science, Department of Computer Science, University of Skövde, Sweden, 1997.

Research Interests

Main research interests is in software testing, in particular testing of real-time systems. Other interests include distributed systems, component technology, and scheduling. Member of the Distributed Real-Time Systems research group since 1998.