

ARTIST 2

Network of Excellence

IST-004527 ARTIST2:
Embedded Systems Design

Activity Progress Report for Year 2

JPIA-Platform Timing Analysis Platform

Clusters:

Compilers and Timing Analysis

Activity Leader:

Reinhard Wilhelm (Saarland University)
<http://rw4.cs.uni-sb.de/users/wilhelm/wilhelm.html>

Policy Objective (abstract)

Combine the best components of existing European Timing-Analysis tools and prototypes in a standard tool architecture with well-defined textual interfaces. Our objective is to integrate European efforts on the Timing Analysis of Real-Time Systems, to preserve the existing lead of European Research and Industry in this important sector.

The resulting platform will be used in teaching the technology all over Europe.

Table of Contents

1. Overview of the Activity	3
1.1 ARTIST2 Participants: Expertise and Roles	3
1.2 Affiliated Participants: Expertise and Roles	3
1.3 Starting Date, and Expected Ending Date	3
1.4 Baseline	4
1.5 Problem Tackled in Year2	4
1.6 Comments from Previous Review	5
1.6.1 <i>Reviewers' Comments</i>	5
1.6.2 <i>How These Have Been Addressed</i>	5
2. Summary of Activity Progress	6
2.1 Previous Work	6
2.1.1 <i>Work achieved in the first 6 months</i>	6
2.1.2 <i>Work achieved in months 6-12</i>	7
2.2 Current Results	8
2.2.1 <i>Technical Achievements / Outcomes / Difficulties encountered</i>	8
2.2.2 <i>Publications Resulting from these Achievements</i>	10
2.2.3 <i>Keynotes, Workshops, Tutorials</i>	11
3. Future Work and Evolution	14
3.1 Problem to be tackled over the next 18 months (Sept 2006 – Feb 2008)	14
3.2 Current and Future Milestones	14
3.3 Indicators for Integration	15
3.4 Main Funding	16
3.5 Internal Reviewers	16

1. Overview of the Activity

1.1 ARTIST2 Participants: Expertise and Roles

Prof. Dr. Reinhard Wilhelm – Saarland University (Germany)

Activity Leader, Compiler design, Static Program Analysis, Timing Analysis

Saarland University has developed much of the timing-analysis technology that is further developed and commercialised by its spin-off company AbsInt. Saarland University is coordinating the integration activity.

Dr. Christian Ferdinand – AbsInt GmbH (Germany)

Leading Tool Supplier

AbsInt provides advanced WCET analysis tools for a wide variety of targets. The tools are used mainly in the avionics and automotive domain. The work within Artist 2 focuses on the advance of WCET analysis techniques by providing and defining interchange formats for the components of WCET tools like AIR (Artist 2 Intermediate Program Representation for WCET tools). This will allow better and closer cooperation with the various research groups in Europe.

Dr. Guillem Bernat – University of York (UK)

Tool Supplier

The contribution of University of York is on measurement-based WCET analysis where its focus is on instrumentation methods, coverage analysis, and timing documentation.

Prof. Dr. Björn Lisper – Mälardalen university (Sweden)

Timing analysis tools

Mälardalen University is working on automatic flow analysis, WCET-analysis case studies on industrial code, the maintenance of a WCET-benchmark suite, the definition of interface formats for timing analysis, and the use of WCET tools in education.

Dr. Peter Puschner – TU Vienna (Austria)

Timing-Analysis Tools and temporally predictable HW-SW architectures

Within the Timing-Analysis Platform Activity TU Vienna focuses on the test-data generation for measurement-based WCET analysis, the definition of the interchange representation for WCET-related information, and on time-predictable architectures.

Dr. Niklas Holsti – Tidorum Ltd. (Finland)

Timing-Analysis Tools

Tidorum Ltd supplies the timing analysis tool Bound-T. Tidorum takes part in the definition of the architecture of the tool platform and in particular in the definition of the interchange representation, AIR. Later, Tidorum will integrate Bound-T with the platform by adding AIR export and import functions.

1.2 Affiliated Participants: Expertise and Roles

Dr. Isabelle Puaut – IRISA (France)

Timing-Analysis Tools

1.3 Starting Date, and Expected Ending Date

September 1st, 2004 until the European timing-analysis platform has been constructed.

1.4 Baseline

Europe is leading this field. The only commercially available WCET tools, aiT¹, Bound-T², and RapiTime³, and most of the academic prototypes, Heptane⁴, SymTAP⁵, SWEET⁶, and the tools of TU Vienna⁷, are of European origin.

The commercial tools and the academic prototypes follow different approaches, namely analytical and measurement-based approaches. Each individual approach has to solve essentially the same set of sub-problems. The differences between the approaches lie in the methods used to solve some of the subproblems. Methods for some subproblems can be combined across approaches.

The timing-analysis problem has been essentially solved. Projects for the development of time-critical systems in industry use tools of ARTIST2 partners. However, there is room for improvement. The usability can be improved by reducing the necessary amount of user interaction. The efficiency of the tools can be improved by improving our knowledge about the underlying processor architecture. The tool realization can be simplified by computer support. The different tools offer different strengths in the areas of different subproblems. The goal is to combine the respective strengths.

Proposals for a modular framework allowing the integration of prototypical developments existed before the start of the NoE. Documented and supported interfaces existed for the individual tools, but not for any combination of components from different tools.

In an ongoing discussion between the participants about WCET-tool architectures, interfaces, and integration agreement on interfaces has been reached.

1.5 Problem Tackled in Year2

The consortium has agreed on a common interchange format, AIR, a frozen version of AbsInt's CRL2 interface specification language. This is the most important advance for the integration activity. AbsInt has produced a syntax specification of AIR.

USaar has produced a proposal for a computation semantics after discussion with Mälardalen and Tidorum.

Mälardalen has set up and maintains set of benchmarks for WCET tools on their web-page and has created guidelines for future benchmark contributions.

The consortium, together with several external experts, has created rules for a WCET tool challenge and invited tool providers to this challenge.

TU Vienna together with Mälardalen has performed a study for a common annotation language which could be used for WCET analysis.

Mälardalen used WCET-Tools (Bound-T) in Real-time systems education as a part of JPASE.

¹ <http://www.absint.com/ait/>

² <http://www.tidorum.fi/bound-t/>

³ <http://www.rapitasystems.com/wcet.html>

⁴ <http://www.irisa.fr/aces/work/heptane-demo/heptane.html>

⁵ <http://www.ida.ing.tu-bs.de/projects/symta/home.e.shtml>

⁶ <http://www.mrtc.mdh.se/projects/wcet/>

⁷ <http://www.wcet.at/>

Parametric Timing Analysis has been jointly elaborated at Saarland University and Mälardalen. TU Vienna and University of York have investigated the problems that are specific to measurement-based WCET analysis methods. In particular they have started to work towards coverage criteria and metrics that are suitable for a measurement-based WCET assessment.

1.6 Comments from Previous Review

1.6.1 Reviewers' Comments

This task is focused on combining the best components of existing European Timing-Analysis tools and prototypes in a standard tool architecture with well-defined textual interfaces. The primary effort for Y1 was to agree standard tool architecture and a set of textual interfaces.

The deliverable documents high-level bullets of outcomes; the presentation at the review expanded upon these bullets to better document the outcomes.

The revised document incorporates some more outcomes shown in the presentation. A limited separation between the foreground (integrative) outcomes from the background outcomes is now achieved.

However, only minor effort was done to enhance the quality of the deliverable. The introduced picture is unreadable – please correct this. More care should be exercised on checking the quality of the document.

The revised document is accepted for the time being, but it needs to be updated.

1.6.2 How These Have Been Addressed

The report clearly separated background material (Section 2.4 and 2.5) from foreground material (Section 3.1. and 3.2).

The figure was destroyed in the conversion from Word to PDF after the combination of the individual contributions to the complete report. This was out of control of the authors.

The problematic figure was replaced by a table.

2. Summary of Activity Progress

2.1 Previous Work

2.1.1 Work achieved in the first 6 months

We started with an extensive discussion of potential tool-interface languages. Such a language needs to offer the representation of object programs with an adequate attribute mechanism to store annotations about feasible and infeasible paths for program execution and to present analysis results at the program-source level in sufficient detail to be easily understandable by the tool user.

The prime candidate for such a language was CRL2, the interface language of AbsInt's timing-analysis tool aiT. CRL2 is the result of long evolution of intermediate representations in analysis frameworks of AbsInt and Saarland University.

The commercially-supported CRL2 format was interfaced with several of the components needed for the planned Timing-Analysis platform, i.e., parts of the analysis tool chain made ready to communicate via CRL2. With the selection of CRL2 a robust and generic interface was introduced into the ARTIST2 framework.

CLR2 is a generic and processor independent format usable for static analysis (including WCET analysis), optimisation of machine code and assembly language. It supports the integrated representation of control flow graph and intermediate analysis results. An efficient C/C++ library reads/writes CRL2 interface files in a text-representation format and provides an API to the data structures used by the components of the timing-analysis tool suite.

CRL2 has an interface to the Program-Analyzer Generator developed at Saarland University, such that interprocedural analyses are easily implemented.

For the given reasons, CRL2 was first chosen for the integration of various work groups' analyses. Several partners in the Timing-Analysis cluster started to interface with CRL2. Experiments at IRISA and Tidorum with the CRL2 library of AbsInt showed good results. In particular, preliminary experiments made at IRISA used CRL2 to add a tree structure on top of the control flow graph supported CRL2 library. The experiments showed that the CRL2 library, through its attribute system, is flexible enough to define data structures used by WCET analysis methods different from those implemented in aiT.

Based on the first successful results with CRL2 it was discussed that the interchange format should not only be used for timing analysis but should also serve as a compiler-analyser interface, thus facilitating the WCET-aware compilation of code. The interface language is called AIR, for Artist 2 Intermediate Representation. AIR is based on CRL2. CRL2 is a dialect (superset) of AIR as explained in section 2.2.1. Further, it was found that in order to suit the needs of the partners of the CTA cluster, a text format of the interchange format had to be defined and the documentation should be extended (until then CRL2 was available as a library only). AbsInt offered to provide this format definition and documentation.

Besides the work on the common interchange format for timing analysis, the partners of the CTA cluster accomplished the following:

- Students from Mälardalen University performed a number of industrial WCET-analysis case studies in Swedish enterprises, mostly with AbsInt's tool aiT. The cases studies showed that for common embedded processors good WCET bounds could be obtained. However, the effort needed to provide the required program-flow information by hand turned out to be high and time consuming.

- Mälardalen and Vienna developed a prototype editor plug-in that helps the programmer to achieve more predictable timing and aims at simplifying WCET analysis. By highlighting code segments that are executed conditionally, i.e., that are executed only for a subset of all possible inputs to a piece of code, the editor helps the programmer identify possible sources of timing variability. Avoiding such input-dependent conditionals, or at least minimising them in length, leads to code that in general has fewer execution paths, a smaller execution-time jitter, and is thus easier to analyse for its WCET.

2.1.2 Work achieved in months 6-12

The major focus of the activity was on the further development of the common interchange format for WCET analysis:

We identified one important area that was not explicitly represented in CRL2: the “instruction set semantics”, by which we mean the computation that is performed by each instruction in the program. As CRL2 has been used so far, the control-flow graph identifies the instructions and their operands, but an analysis tool must itself have enough knowledge of the target processor to map the instruction identifier and operand identifiers to the computation, for example to understand that the instruction adds two registers and puts the sum in a third register. This computation, connected to the control-flow graph, is the essential information that the tools from Mälardalen and Tidorum use to find constraints on execution flow, such as loop bounds. We therefore decided to define an explicit and general representation of this instruction set semantics as one part of AIR, using the flexible attribute mechanism from CRL2. Work on this extension is under way.

Peter Marwedel's group at Dortmund University used CRL2 to interface their compiler with the aiT tool. Similarly to earlier activities, one of the results was that a better documentation on aiT's usage of attributes would be useful.

While AbsInt and Saarland University were working on extensions and improvements of CRL2, feedback from Rennes, Mälardalen and Tidorum asked for a formalisation of the control flow graph structure and some further extensions of the interchange format.

In parallel, Vienna and Mälardalen started work on extending the path-annotation support of CRL2. As for the instruction-set semantics, the plan was to use the attribute mechanism of CRL2 to represent these annotations about feasible and infeasible execution paths to facilitate a highly accurate modelling of the possible execution paths for WCET analysis. Inputs from Vienna, Mälardalen, Tidorum, and AbsInt for these path annotations had been collected and summarized in a presentation.

Though each of the activities was very promising by itself, the consortium had to find that the definition of such a complex interface as the WCET interchange format requires and would yet require a lot of further work till its final completion.

Besides the core work on the common format the partners of this activity could report about the following work related to timing analysis:

- Tidorum and Mälardalen cooperated on implementing a version of the Bound-T WCET analysis tool for the Renesas H8/300 processor, which can be found in the popular Lego Mindstorms kit. Mälardalen is now using this tool in real-time systems education.
- Licensing policies and the issue of securing the availability of AbsInt's CRL2 library had been discussed. We are looking for a full and open definition of the syntax and semantics of AIR so that anyone can build their own AIR libraries.
- Vienna and York started to cooperate on measurement-based timing analysis. The partners had been working individually on measurement-based analysis before. York

uses the measurement-based WCET analysis for non safety-critical applications and has a strong focus on code instrumentation and report formats for representing measurement details to the user. In Vienna, the use of measurements is seen as a complement to static analysis for the purpose of validating static-analysis results. In Vienna, the focus is on the automatic test-data generation. The cooperation started within the CTA cluster aims at combining the efforts and exchanging the complementary know-how of the groups. In particular, the partners started to work on the definition of coverage criteria for measurement-based WCET analysis.

- The industrial WCET case studies performed by students from Mälardalen University in Swedish enterprises were continued. The findings essentially confirmed the conclusions from the earlier studies.
- Representatives for AbsInt, Tidorum and Mälardalen all demonstrated their WCET tools and participated in the Real-Time in Sweden (RTiS) conference, held in Skövde, Sweden, Aug 2005. This was a joint effort to introduce the concept of static timing analysis to the Swedish companies that participated in the RTiS conference. See: <http://www.snart.org> for details about RTiS.

A number of publications have been produced as a result of the cooperation inside the cluster. They are included in the list of publications collected and submitted by the coordinator.

2.2 Current Results

2.2.1 Technical Achievements / Outcomes / Difficulties encountered

Definition of AIR (Artist2 Intermediate program representation for WCET tools).

In the past few months, a file format specification was developed to define the AIR ('ARTIST2 Interchange') format that may be used by the cluster participants' tools for integration. So AIR is the proposed exchange format of the tools of the groups participating in the ARTIST2 project. The format is based on CRL2, which is the successor of CRL. These formats were originally developed in cooperation by Saarland University and AbsInt Angewandte Informatik GmbH over several years of work.

The idea behind AIR is that an interface is to be defined on the file-format level, in contrast to CRL2, whose interface definition only covers the C++ library interface. Internally in AbsInt tools, a specification of the C++ library interface is preferred over a file format specification, simply because all tools use the library and thus the storage on disk is secondary. For ARTIST2, different work groups prefer their own libraries over the usage of proprietary software, so there is a serious demand for a file format specification.

Since CRL2 was not primarily meant to be a file format, much work had to be done before this document could be written. Apart from the mere documentation the file format had to be defined and implemented. In order to get a stable interface on file level CRL2 had to be extended. For example, version numbers and specification IDs had to be added to meet the strict safety criteria of real-time systems analysis. Thus, this document can be viewed as the first step of the final documentation phase in a larger effort towards an exchange file-format for the different WCET tools and tool components used within this ARTIST2 activity.

From the release of the first AIR specification on, the CRL2's file format interface will be a dialect of the AIR file format. CRL2 as well as dialects of other work groups are allowed to feature extensions as long as they are not vital for the operation of the tools. E.g., AbsInt tools will only use the plain AIR file format during normal operation, the extensions of CRL2 are mainly implemented for debugging and diagnosis purposes. In the same way, extensions of

other dialects shall never be vital to the operation of the corresponding tools

<http://www.absint.com/artist2/doc/crl2/air.pdf>

Timing-Analysis Survey paper:

The work on the survey paper about Timing-Analysis Methods and Tools has clarified the characteristics, the advantages and disadvantages, and the application domains for the different approaches, e.g. analytical and measurement-based approaches. It has clarified the modularisation of the overall timing-analysis task and the possibility of combining modules for different subtasks across the approaches. The joint authorship of this paper expresses strong and enduring cooperation between the different groups. This paper will represent a landmark publication for the area!

Timing Anomalies Characterisation and Checking

Timing Anomalies in processors produce counter-intuitive timing behaviour, i.e., local worst-case behaviour does not necessarily lead to global worst-case behaviour. The existence of timing anomalies requires complex timing-analysis procedures. Saarland University together with Freiburg University have worked on the clarification of the concept and the origins of timing anomalies. The goal is an automatically checkable definition of timing anomalies, which would allow for a safe reduction of the WCET-analysis effort whenever the absence of timing anomalies can be shown for a processor platform. Furthermore, it was found that certain cache-replacement strategies lead to Domino Effects, which are timing anomalies without bounds for their effects. This work is funded by the Transregional Research Centre AVACS (Automatic Verification and Analysis of Complex Systems) of the Deutsche Forschungsgemeinschaft.

Parametric WCET Analysis.

The runtime of programs might depend on parameters. In these cases the worst case execution time (WCET) has to be recomputed for each parameter assignment. This can be very time consuming. On the other hand the relation between parameters and WCET cannot be easily identified.

Saarland University together with Mälardalen University have initiated collaboration about this type of parametric WCET analysis. Two M.Sc. students from Saarland visited Mälardalen in early 2006 to learn about Mälardalen approach.

In the joint approach we perform a parametric WCET analysis based on the aiT-toolchain. It computes a WCET formula instead of a concrete value. Since programs spend most of their runtime in loops, we focus on a parametric loop-bound analysis. Prior to this part the parameters of the executable have to be determined. In the path analysis part, a parametric optimisation method is needed. Afterwards the resulting formula has to be evaluated. Evaluation in this context means visualisation or instantiation of the formula.

WCET analysis benchmark suite

Mälardalen University has collected a suite of benchmark programs for WCET analysis. The suite is maintained on the web by Mälardalen. One of the purposes of this benchmark suite is to be able to evaluate and compare different WCET tools as is done in the initiated WCET Tool Challenge (cf. <http://www.mrtc.mdh.se/projects/wcet/benchmarks.html>)

Input format for flow analysis

Mälardalen University has initiated work to define a standard input code format "ALF" for flow analysis. The purpose is to facilitate flow analysis of codes in different formats by translating to ALF. ALF will provide an interface to Mälardalen's flow analysis. A first draft for ALF exists, and it will be disseminated within the cluster before the format is finalised. ALF should also be harmonised with the AIR instruction semantics format.

Synergy between Code Synthesis and Timing Analysis

Saarland University together with AbsInt and ETAS have integrated the ASCET-SD code-

synthesis with AbsInt's timing-analysis tool to improve usability of the timing-analysis tool and precision of the results.

Timing Predictability

First quantitative results have been obtained on the influence of architectural properties on the timing predictability of embedded systems. In particular, four different cache replacement policies and their influence on predictability have been considered at Saarland University. This research is funded by AVACS. On the side of TU Vienna, a model for a time-predictable processing node has been worked out – on this node software timing behaviour can be predicted with the granularity of the CPU clock. This node uses a purely time-triggered input-output interface and relies on single-path code (code that is free from input-data dependent control flow) in both the operating system and the application code. Tasks are only preempted at pre-planned task preemption points and simple clock synchronization keeps the operations of the nodes in synchrony with its real-time environment. The work on the time-predictable node yielded a time-predictable task-preemption model where an instruction counter instead of the CPU clock is used to implement preemptions (It was shown that CPU-clock based preemption may lead to unpredictable timing).

Measurement-Based WCET Analysis

As the complexity of WCET analysis varies with the structure of the program to be analysed and the type of target hardware, TU Vienna and York worked out a detailed list of issues for measurement-based WCET analysis. This list of issues is to be used for different purposes: First, it is a check list for the designer of a WCET analysis tool. Second, it gives the system developer clues about relevant hardware and software criteria when designing a system with the goal of simple analysability and predictability. The list is divided into three categories: a) issues that only relate to the software of the system, b) issues that address only the target hardware of the system, and c) issues that are relevant for both, the software and the hardware part of the system. The partners used these results as a starting point for the work on coverage criteria for measurement-based WCET analysis that was initiated in this work period. The goal of this work item is to find meaningful metrics for assessing the timing-related code coverage and the value of input-data sets for the measurement-based analysis.

The results are documented in a technical report, which is available at:

<http://www.vmars.tuwien.ac.at/php/pserver/docdetail.php?DID=1975&viewmode=paper&year=2006>

2.2.2 Publications Resulting from these Achievements

Jochen Eisinger, Ilia Polian, Bernd Becker, Alexander Metzner, Stephan Thesing, Reinhard Wilhelm: *Automatic Identification of Timing Anomalies for Cycle-Accurate Worst-Case Execution Time Analysis*. DDECS 2006: 15-20

Jan Reineke, Bjoern Wachter, Stephan Thesing, Reinhard Wilhelm, Ilia Polian, Jochen Eisinger, and Bernd Becker: *A Definition and Classification of Timing Anomalies*, WCET 2006, Dresden, <http://moss.csc.ncsu.edu/~mueller/wcet06/wcet06.pdf>

Christoph Berg: *PLRU Cache Domino Effects*, WCET 2006, Dresden, <http://moss.csc.ncsu.edu/~mueller/wcet06/wcet06.pdf>

Jan Reineke, Daniel Grund, Christoph Berg, Reinhard Wilhelm: *Predictability of Cache Replacement Policies*, AVACS Technical Report 9, 2006

Christian Ferdinand, Reinhold Heckmann, Hans-Joerg Wolff, Christian Renz, Oleg Parshin, Reinhard Wilhelm: *Towards Model-Driven Development of Hard Real-Time Systems – Integrating ASCET-MD and aiT/StackAnalyzer* –, Advanced Automotive Software and Systems Development: Model-Driven Development of Reliable Automotive Services, San Diego, CA (USA), March 15-17, 2006

Daniel Sehlberg, Andreas Ermedahl, Jan Gustafsson, Björn Lisper, and Steffen Wiegratz. *Static WCET Analysis of Real-Time Task-Oriented Code in Vehicle Control Systems*. Accepted for publication at ISoLA-06, Cyprus, Nov. 2006.

Jan Gustafsson, Björn Lisper, Raimund Kirner, Peter Puschner, *Code Analysis for Temporal Predictability, Real-Time Systems*, vol 32, no 3, pp. 253 - 277, Springer-Verlag, March, 2006

D. Kazakov and I. Bate. "Towards new methods for developing real-time systems: Automatically deriving loop bounds using machine learning". To Appear in Proceedings of the 11th IEEE International Conference on Emerging Technologies and Factory Automation, 2006.

A. Betts, G. Bernat. "Tree-Based WCET Analysis on Instrumentation Point Graphs" (2006) Proceedings of the 9th IEEE International Symposium on Object-Oriented Real-Time Distributed Computing (ISORC)

R. Kirner, I. Wenzel, B. Rieder, P. Puschner. *Using Measurements as a Complement to Static Worst-Case Execution-Time Analysis*, in Intelligent Systems at the Service of Mankind Vol. 2, pp. 205-226, UBooks, Jan. 2006, ISBN 3-86608-052-2.

R. Kirner, P. Puschner, I. Wenzel, B. Rieder. *Portable Data Exchange for Remote-Testing Frameworks*. In Proceedings of the 9th IEEE International Symposium on Object- and Component Oriented Real-Time Distributed Computing, pp. 476-482, April 2006.

P. Puschner, R. Kirner. *From Time-Triggered to Time-Deterministic Real-Time Systems*. In Proceedings 5th IFIP Working Conference on Distributed and Parallel Embedded Systems, Oct. 2006.

R. Kirner, M. Grössing, P. Puschner. *Comparing WCET and Resource Demands of Trigonometric Functions Implemented as Iterative Calculations vs. Table-Lookup*. In Proceedings 6th International Workshop on Worst-Case Execution Time Analysis, July 2006.

G. Bernat, A. Betts, R. Kirner, P. Puschner. *Software and Hardware Issues of Measurement-Based Worst-Case Execution Time Analysis*. Technical Report 72/2006, Vienna University of Technology, Institute of Computer Engineering, 2006.

Reinhard Wilhelm, Jakob Engblom, Andreas Ermedahl, Niklas Holsti, Stephan Thesing, David Whalley, Guillem Bernat, Christian Ferdinand, Reinhold Heckmann, Frank Mueller, Isabelle Puaut, Peter Puschner, Jan Staschulat, and Per Stenström. *The Worst-Case Execution Time Problem - Overview of Methods and Survey of Tools*, under revision for ACM Transactions on Embedded Computing Systems.

2.2.3 Keynotes, Workshops, Tutorials

ARTIST2 Summer School: Component & Modelling, Testing & Verification, and Static Analysis of Embedded Systems

Nässlingen, Sweden – Sept, 29th – Oct, 2nd, 2005

The ARTIST2 Summer School was a 4 day summer school for young researchers working or wanting to work in the fields of modelling, validation and performance analysis of embedded systems as well as engineers from industry with practical background in design and testing of embedded systems. The school was completely booked out, mainly with young people from all over Europe.

<http://www.artist-embedded.org/FP6/ARTIST2Events/SummerSchools/Artist05.html>

Workshop: 6th Intl WORKSHOP ON WORST-CASE EXECUTION TIME ANALYSIS 18th Euromicro Intl Conference on Real-Time Systems

Dresden, Germany – July 4, 2006

The goal of the workshop was to bring together people from academia, tool vendors, and users in industry, which are interested in all aspects of timing analysis for real-time systems. The workshop has provided a relaxed forum to present and discuss new ideas, new research directions, and to review current trends in this area. The workshop was based on short presentations that have encouraged discussion by the attendees.

The topics of the workshop included any issue related to timing analysis, in particular:

- Different approaches at computing WCET
- Flow analysis for WCET
- Low-level timing analysis, modelling and analysis of processor features
- Calculation methods for WCET
- Strategies to reduce the complexity of WCET analysis
- Integration of WCET and schedulability analysis
- Evaluation and case studies
- Testing Methods for WCET analysis
- Tools for timing analysis
- Design for Timing Predictability
- Integration of WCET analysis into the development process
- Compiler optimizations for worst-case paths
- WCET analysis for multi-processors, multi-cores or SMTs
- WCET analysis for networks (e.g., CAN)

<http://moss.csc.ncsu.edu/~mueller/wcet06/>

Platform Meeting: Timing Analysis Platform meeting

Amsterdam, Netherlands – November 8, 2005

Agenda:

- Status of Activities
- Work plan for next 18 months
- Spreading Excellence
- Presentations

Platform Meeting: Timing Analysis Platform meeting

Munich, Germany – March 7, 2006

Agenda:

- Report of the follow-up review, 27.1.06 Brussels, and the Strategic Management Board meeting, 28.2.06 Paris
- AIR (ARTIST2 Intermediate Representation)
 - Specification (AbsInt)
 - Representation of ISA Semantics (Saarland University)
- Flow representation
 - NIC- - (MdU)
 - Flow attributes(TU Vienna)
- WCET benchmark collection of Mälardalen
- ESA project of Tidorum and Rapita
- Coupling between Dortmund compiler backend and aiT

Platform Meeting: Timing Analysis Platform meeting*Dresden, Germany – July 5, 2006*

Agenda:

- Instruction semantics (computation semantics) and their representation in AIR (Stephan Thesing)
- Presentation of AIR (AbsInt)
- Reporting for the ARTIST2 Review
- Adapting SWEET to CRL2 (Mälardalen)
- The ISoLA special track
- Discussion of the WCET Tool Competition

Special Track on Timing Analysis in the Industrial Development Process**2nd International Symposium on Leveraging Applications of Formal Methods, Verification and Validation (ISoLA) sponsored by ARTIST2.***Paphos, Cyprus – November 15 – 19, 2006*<http://sttt.cs.uni-dortmund.de/isola2006/index.php?id=tracksandsession#a2>

Many safety-critical embedded systems have to satisfy hard real-time constraints. These systems need sound methods and tools to derive reliable run-time guarantees. The guaranteed run times should not only be reliable, but also precise. The achievable precision is highly dependent on characteristics of the target architecture and on the software design and implementation method.

Experience has shown that a tight integration of Timing Analysis into the development process and the development tool chain improves the achievable precision. This Special Track will be concerned with questions around the integration of timing analysis in the industrial development process.

Topics of interest:

- Positioning Timing Analysis in the development process,
- implications for system design and implementation,
- timing predictability of embedded systems,
- validation of timing-analysis methods and tools,
- Degrees of time-criticality and corresponding methods.
- The results from the Timing-Analysis Tool Challenge 2006 will be presented.

3. Future Work and Evolution

3.1 Problem to be tackled over the next 18 months (Sept 2006 – Feb 2008)

Most of the work will concern the basis for tool integration, the chosen interface language AIR and its extensions for different purposes.

1. **AIR Semantics**

The semantics of the chosen interface language will be specified.

2. **Computation semantics language**

- a. Discussion of the computation semantics language as an extension of AIR.
- b. Definition of the primitive operations
- c. Specification of the semantics for a core language

3. **ALF flow analysis input format**

The ALF flow analysis input format will be finalized, and Mälardalen's flow analysis will be adapted to use ALF instead of the current NIC format.

4. **The WCET Tool Challenge**

The WCET Tool Challenge will be run in the fall 2006. It is based on a set of WCET benchmarks. Developers are invited to submit their WCET tools to be run against these benchmarks. The purpose is to be able to study, compare and discuss the properties of different WCET tools and approaches, to define common metrics, and to enhance the existing benchmarks. (cf. <http://www.idt.mdh.se/personal/jgn/challenge/>). This is only the first instance of the WCET Challenge, which is intended to be an annual event.

5. **Measurement-Based Analysis**

The work on coverage criteria for the measurement-based analysis will be continued and documented in a report. Further, the prototype framework for test-data generation will be improved.

3.2 Current and Future Milestones

- Year2: Standard tool architecture and interfaces
 - The tool architecture has been further clarified. In the course of writing the joint survey paper, modularity of the architecture has been further extended (*100% completed*).
 - An agreement on the ARTIST2 common interface language, AIR, has been reached. The syntax has been defined, the semantics is accepted, but has not been formally written down (*60% completed by writing up the syntax for the agreed semantics*).
 - The extension of AIR for the computation semantics has been conceived. Basic operations have, yet, to be defined (*50% completed by specifying the approach to the specification of computation semantics*)
- Year3: Initial integration of existing components
 - The chosen interface language, AIR, is being extended by Saarland University and by AbsInt to suit the needs of other partners.
 - Four partners of the team (Vienna, Mälardalen, Tidorum, AbsInt) will continue to work on path description attributes for AIR to arrive at a uniform notation.

- Mälardalen will wrap up its flow analysis into a component with well-defined interfaces, which will be integrated with the aiT tool of AbsInt and the Bound-T tool of Tidorum.
- Year4: Integration of existing components

3.3 Indicators for Integration

In the first two years of the project considerable integration work has taken place:

- The partners have agreed on standard tool architecture and a set of textual interfaces.
- Experiments with interfacing through these textual interfaces have been successfully performed.
- Several professional components of aiT and Bound-T are being offered by the industrial partners to be used by the academic partners.
- AbsInt's aiT tool has been used in industrial case studies by the Mälardalen team (see publications).
- The Bound-T version for Renesas H8/300 is used in real-time education at Mälardalen.
- Tidorum and York (through Rapita Systems Ltd) are working on a project for the European Space Agency, to adapt their timing analysis tools to the LEON2 (SPARC V8) processor and to evaluate the tools on real on-board satellite software. This includes integration between the tools: Tidorum's Bound-T tool will be used as a front-end for York's measurement-based analysis as implemented in the RapiTime tool from Rapita Systems.
- TU Vienna and University of York worked on issues of how systematic measurements can be used for WCET analysis.

Indicators for integration in the future:

- Working integration of Mälardalen's flow analysis with the aiT tool of AbsInt and the Bound-T tool of Tidorum.
- Development and documentation of path description attributes for AIR.
- Joint publications about Timing Analysis.

The following table shows an overview of the cooperation on the timing analysis platform.

	USaar	AbsInt	York	MdU	TUW	Tidorum	IRISA
Definition of common interface format AIR	+	+	+	+	+	+	+
Extension for Instruction Set Semantics	+	+		+		+	
Extension for Path Annotation		+	+	+	+	+	
CRL2 usage experiments	+	+				+	+
Industrial case studies with the aiT		+		+			
Bound-T for the H8/300 and Real-time systems teaching				+		+	
Measurement based timing analysis			+		+		

	USaar	AbsInt	York	MdU	TUW	Tidorum	IRISA
Bound-T as a front-end for RapiTime			+			+	
Integrating the MdU's flow analysis into aiT		+		+			
Integration the MdU's flow analysis into Bound-T				+		+	
Benchmark programs for timing analysis tools			+	+			
Parametric WCET analysis	+			+			
WCET-oriented IDE				+	+		

3.4 Main Funding

Main sources of funding are:

- TU Vienna's work is funded mainly from the DECOS Integrated Project and the MoDECS project (funded by the Austrian FIT-IT programme; duration: Sept. 2003-Aug. 2005), and the Te-DES Project (funded by FIT-IT; duration: Apr. 2005-Mar. 2007) and the COSTA project (funded by the Austrian science funding agency FWF; duration: July 2006-June 2009).
- Saarland University's funding comes from the national project AVACS (DFG).
- Uppsala and Mälardalen had ASTEC funding from VINNOVA until 2005. From 2006, Mälardalen has funding from the KK-foundation (grant 2005/0271, duration Jan 2006 – Dec. 2008), and the Foundation for Strategic Research (PROGRESS Strategic Research Centre, duration Jan. 2006 – Dec. 2010).
- AbsInt currently receives funding from a national (BMBF) project, VERISOFT.
- Work at York has been funded by BaE systems and by NextTTA EU project IST-2001-32111.
- Tidorum's work on integrating their Bound-T tool with the RapiTime tool (originating at the University of York) is funded by the PEAL project for the European Space Agency.

3.5 Internal Reviewers

Philipp Lucas, Saarland University and Peter Puschner, TU Vienna.