ARTIST 2

Network of Excellence

IST-004527 ARTIST2:
Embedded Systems Design

Activity Progress Report for Year 1

JPIA-Platform:

Timing Analysis Platform

Cluster:

Compilers and Timing Analysis

Activity Leader:

Reinhard Wilhelm (Saarland University)

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. Introduction

1.1 Activity Leader

Team Leader: Reinhard Wilhelm (Saarland University)
Areas of his team's expertise: Compiler Design, Static Program Analysis, Timing Analysis, Coordinator.

1.2 Policy Objective

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.

1.3 Industrial Sectors

Timing-Analysis tools have recently entered industrial practice and are in routine use in the aeronautics and automotive industry. Precision of the results is good, efficiency is tolerable, and usability needs improvement.

1.3.1 Possible Global Impacts of Research Results

Experience with Timing-Analysis tools in industry has shown that the awareness of the problem of designing systems with timing-predictable behaviour should be raised, since the trends in system design lead to more and more unpredictability. More speculation in processor architecture, more complex software architectures lead to greater variability in execution times.

The impacts of the research done in the cluster will show how far the combination of the most advanced techniques can lead, and in consequence strengthen the insight that a change in the design methods has to take place.

2. Overview of the Activity

2.1 Artist Participants and roles

Team Leader: Reinhard Wilhelm (Saarland University)
Areas of his team's expertise: Compiler Design, Static Program Analysis, Timing Analysis

Team Leader: Christian Ferdinand (AbsInt)
Areas of his team's expertise: leading tool supplier.

Team Leaders: Iain Bate, Guillem Bernat (York)
Areas of the team's expertise: Timing-Analysis Tools.

Team Leaders: Andreas Ermedahl, Jan Gustafsson, Björn Lisper (Mälardalen)
Areas of the team's expertise: Timing Analysis Tools.

Team Leader: Peter Puschner (TU Vienna)
Areas of his team's expertise: Timing-Analysis Tools and temporally predictable HW-SW architectures.

2.2 Affiliated partners and Roles

Team Leader: Niklas Holsti (Tidorum)
Areas of his team's expertise: Timing-Analysis Tools.

Team Leader: Isabelle Puaut (IRISA)
Areas of his team's expertise: Timing-Analysis Tools.

2.3 Starting date, and expected ending date

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

2.4 Baseline

Europe is leading this field. The only commercially available WCET tools, aiT[1], Bound-T[2], and RapiTime[3], and most of the academic prototypes, Heptane[4], SymTA/P[5], SWEET[6], and the tools of TU Vienna[7], are of European origin. Proposals for a modular framework allowing the integration of prototypical developments and documented, supported interfaces exist.

---

[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/

This is the continuation of an ongoing discussion between the participants about WCET-tool architectures, interfaces, and integration.

2.5 Technical Description

Each partner brings specific competencies to be integrated in the common platform:

Most partners the CTA cluster use the cache-behaviour prediction developed at Saarland.

Uppsala and Mälardalen have developed sophisticated flow-analysis concepts and special-case optimized path-analysis methods. They also developed strong analyses for component-behaviour prediction for simple processors.

AbsInt and Tidorum provide professional tool components for WCET analysis. AbsInt’s aiT tool offers several front ends for different target processors ranging from simple to very complex, high-end processors. Tidorum's Bound-T tool supports different target processors, with a broader focus on simple and more predictable processors.

IRISA imports their tree-based WCET calculation to the common efforts of the cluster.

York has developed tools both for static memory analysis and measurement-based timing analysis. Within the cluster, the measurement-based analysis will establish a complement to static analysis by its use for validating the static analysis and providing an alternative analysis for software that is not safety-critical.

Vienna has developed tools for both static and measurement-based timing analysis, as well as concepts for HW/SW architectures that improve timing predictability. Within the CTA cluster, the contributions are in the area of path description languages for characterizing feasible and infeasible paths for the static WCET analysis and on the test-data generation for the measurement-based analysis.

3. Activity Progress Report

3.1 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 well-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 Universität des Saarlandes.

CRL2 is commercially supported and is interfaced with several of the components needed for the planned Timing-Analysis platform, i.e. parts of the analysis tool chain communicate already via CRL2. With the selection of CRL2 a robust and generic interface is 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 is interfaced with the Program-Analyzer Generator developed at Saarland University, such that interprocedural analyses are easily implemented.

For these reasons, CRL2 was found suitable for 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 tried to use 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.

CRL2 is not only used for timing analysis but is also planned to serve as a compiler-analyser interface, thus facilitating the WCET-aware compilation of code.

Having selected CRL2 as the interface language for the CTA platform, the following steps need to be performed:

To suit the needs of the partners of the CTA cluster the text format of CRL2 must be defined and the documentation extended. This documentation and definition of an external text format will be provided by AbsInt (until then CRL2 is a library).

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

- Several industrial case studies performed by students from Mälardalen 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, much effort was needed to provide the required program flow information by hand.
- Mälardalen and Vienna have developed a prototype editor plugin 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 minimizing 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.

3.2 Work achieved in months 6-12

It turned out that Tidorum and Mälardalen need an explicit representation of instruction-set semantics in CRL2 in order to use the CRL2 interface in the Bound-T tool. The instruction set semantics can be provided in CRL2 by CRL2's flexible attribute mechanism. Work for this extension is under way.

Prof. Marwedel's group at Dortmund University is using CRL2 to interface their compiler with aiT tools. This requires a better documentation of aiT's usage of attributes. The work on this has started.

Feedback from Rennes, Mälardalen and Tidorum asked for a formalisation of the control flow graph structure and some format extensions. The work on extending CRL2 at AbsInt and USaar is still ongoing.

It has been found that the definition of a complex interface requires and will yet require a lot of work.

Vienna and Mälardalen started work on extending the path annotation support of CRL2. Again, the attribute mechanism of CRL2 will be used to represent these annotations about feasible and infeasible execution paths to facilitate a highly accurate modelling of the possible execution paths for WCET analysis. So far, inputs from Vienna, Mälardalen, Tidorum, and AbsInt for these path annotations have been collected. A first evaluation has been documented in a slide presentation.

Further work related to timing analysis by partners of this activity:

- 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 now uses this tool in real-time systems education.
- Discussion of licensing policies and securing of the availability of AbsInt's CRL2 library. No final solution is found here yet.
- Vienna and York initiated cooperation 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, cooperative work on defining coverage criteria for measurement-based WCET had been started. A report on first results is currently under preparation.
- Still more industrial case studies were performed by students from Mälardalen in Swedish enterprises. 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.

3.3 Difficulties Encountered

We encountered the normative power of facts, i.e. existing implementation of an interface language; some other partners would have liked to switch from CRL2 to an XML-based format, but due to the limited resources available within the project it was decided to do without a re-definition and reimplementation.

The start was slow because the only candidate tool-interface language CRL2 of AbsInt lacked documentation. Meanwhile AbsInt gave some information support on CRL2 and is working on a documentation that will be provided to the partners.

Interests of participants changed, e.g. for the Predictability issue and for the Integration-with-Compiler issue (this activity was cancelled during contract negotiations).

A critical property of the funding instrument, NoE, is that outside funding determines interests and orientation!

3.4 Recommendations

Tidorum should become a full partner as Niklas Holsti is very engaged in the cluster's activities.

3.5 Milestones

AbsInt's tool has been used in the development of time-critical systems in the Airbus A380.

3.6 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).
- Saarland University's funding comes from the national project AVACS (DFG).
- Uppsala and Mälardalen have ASTEC funding from VINNOVA
- 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.

3.7 Indicators for Integration

In the first year 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.
- The chosen interface language is being extended by Saarland University and by AbsInt to suit the needs of other partners.
- 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.

Indicators for integration in the future:

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.

A paper on programming support for WCET-oriented programming co-authored by Mälardalen and Vienna will be published.

Four partners of the team (Vienna, Mälardalen, Tidorum, AbsInt) have started work on a path description attributes for CRL2. This work will be further pursued and documented.

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

	USaar	***AbsInt***	***York***	***MdU***	***TUW***	***Tidorum***	***IRISA***
jointly	1	1, 8	1	1, 8	1	1, 8	1
USaar		2		2		2, 4	
AbsInt				5, 9	3	4	4
York				11	3, 7		
MdU					3	6, 10	
TUW						3	
Tidorum							

1. CRL2 definition as a common interface format.
2. CRL2 extension for Instruction Set semantics.
3. CRL2 extensions for path annotation.
4. CRL2 usage experiments.
5. Industrial case studies with the AbsInt aiT tool.
6. Bound-T for the H8/300 and real-time systems teaching.
7. Measurement-based timing analysis.
8. Tool demonstrations (RTiS 2005).
9. Integrating the MdU flow analysis into aiT.
10. Integrating the MdU flow analysis into Bound-T.

11. Benchmark programs for timing analysis tools.

3.8 Evolution

The interface language will be finalized during the next 6 months. USaar will implement a way to store the instruction-set semantics into CRL2 as needed by Tidorum and Mälardalen.

Integration of existing components will be done after that. New components developed in accompanying projects will continue to be integrated.

A new activity is the integration of a compiler with a timing-analysis tool to obtain more information about program flow. This activity was originally planned but cancelled to reduce the number of activities. However, due to recent interest at a compiler company, this activity may be revitalized.

USaar has started cooperation about Design for Predictability with partners in the Execution-Platform cluster, ETHZ, Bologna, and Dortmund.

Mälardalen together with York will establish a benchmark for time-critical programs to evaluate timing-analysis tools.

Mälardalen will package its flow-analysis methods into a program component, which can be integrated with other tools.

York will cooperate with TU Vienna on measurement-based timing-analysis methods.

Vienna, Mälardalen, AbsInt, and Tidorum will continue their work on the path-description attributes for CRL2.

4. Detailed Technical View

4.1 Brief State of the Art

Hard real-time systems need to satisfy stringent timing constraints, which are derived from the systems they control. A schedulability analysis for the set of tasks making up the system and a given hardware architecture has to be performed in order to guarantee that all the timing constraints of these tasks will be met (timing validation). Schedulability analysis requires knowledge of upper bounds for the execution times of all the system's tasks. These upper bounds (and lower bounds) have to be safe, i.e., they must never underestimate (overestimate) the real execution time. The upper bounds represent the worst-case guarantee the developer can give.

Furthermore, the bounds should be tight, i.e., the overestimation (underestimation) should be as small as possible. Thus, the main criteria for the acceptance of a timing-analysis tool that is used to obtain guarantees are soundness of the methods-do they produce safe results?-and precision-are the derived bounds tight?

The problem of determining upper bounds on execution times for single tasks and for quite complex processor architectures has been solved. Several commercial WCET tools are available. They have experienced positive feedback from extensive industrial use in the automotive and aeronautics industry. This feedback concerned performance of the tools, precision of the results, and usability. In addition, several research prototypes are under development.

Timing-Analysis tools have to cover a rather large space originating from different application domains with their specific requirements, different classes of processor architectures, more general hardware and overall system architectures, and different user expectations. The tools offered by the cluster partners have their strengths in different points of this space.

The existing tools serve some particular and highly relevant points in this space. AbsInt's tool for example has been used in the development of time-critical systems in the Airbus A380.

On the other hand, they currently do not serve distributed architectures well. Another interesting point in this space is occupied by timing-analysis tools adapted for teaching. This is approached by cooperation between Mälardalen and Tidorum.

The tool development is highly complex connected to the high complexity of modern processor architectures. It takes too much effort and it is error prone. Therefore, computer-supported tool-development efforts are needed.

4.2 Industrial Needs and Experience

The commercially available tools, aiT by Absint, Bound-T by Tidorum, and RapiTime by Rapita Systems, have been tested both in industrial projects in the automotive and aeronautics industries and in academia-industry cooperations.

Reports about experience have been published in several conferences and workshops. Some potential industrial customers expect timing-analysis tools to do miracles. They expect that an a-posteriori analysis of an arbitrarily designed, implemented, composed system on top of arbitrarily complex hardware architecture could be arbitrarily precise. This is definitely not realistic! Best results are achieved when the timing-analysis technology is incorporated into the design process.

Further precision could be obtained if a better integration of timing-analysis in the whole tool chain, in particular with code synthesis, compiler, were realized.

The studies done by Mälardalen have shown that parametric analyses should be realized by timing-analysis tools. These studies have been done on time-critical code in operating systems and code for communication control in vehicles. These codes have a highly parametric timing behaviour.

4.3 Ongoing Work in the Partner Institutions

Work at Saarland University concentrates in the tool-development process. This highly complex process should be supported by computer-supported transformations starting with formally specified processor models. In addition, research has started about the question of how to design systems with a highly predictable timing behaviour.

AbsInt extends its range of supported hardware platforms as required by industrial customers. It has also strengthened the Value Analysis needed for flow analysis and for data cache analysis.

In the Cluster-Integration Activity, AbsInt offers the only realistic interface language for platform integration, CRL2. This language has been experimented with by Tidorum and Rennes with good results. It is being extended to support representation of instruction-set semantics in order to accommodate Tidorum's and Mälardalen's requirements.

Mälardalen has been successful in convincing a compiler manufacturer to support the very attractive idea of the integration of timing analysis with compilation. They have done intensive industrial case studies in students' projects and collected valuable experience. Tidorum and Mälardalen have agreed on to integrate Mälardalen's flow analysis ("abstract execution") into Bound-T to be an alternative of addition to Bound-T's own flow analysis, at the user's option. This integration will need a representation of instruction-set semantics.

Flow analysis of programs can increase significantly the precision of a subsequent timing analysis. Any infeasible path excluded by flow analysis will not contribute any time to the global execution time estimate. Tidorum has realized a powerful flow analysis based on Pugh's Omega test. Mälardalen has developed strong source-level flow analysis concepts. TU Vienna has concentrated on transforming flow facts in lock-step with optimizing transformations.

Rennes has extended its structure-based approach to deal with branch prediction.

4.4 Interaction, Building Excellence Between Partners

The different tools of the partners have different strengths and follow different orientations. Synergies between the research in Mälardalen on flow analysis and the most advanced microarchitecture analysis methods of USaar and AbsInt are promising lines of development. The same holds for the Bound-T tool of Tidorum.

TU Vienna and Mälardalen cooperate in the area of flow analysis and compiler integration.

Mälardalen and AbsInt cooperate in industrial experiments.

Mälardalen and Bound-T implemented a system based on Bound-T for teaching purposes.

Mälardalen together with York will establish a benchmark for time-critical programs to evaluate timing-analysis tools.

The Timing-Analysis Platform activity would not have happened without ARTIST2. Europe's groups were leading when ARTIST2 started – and they still are, all commercially available tools come from Europe. ARTIST2 sponsored the WCET 2005 workshop (chair R. Wilhelm).

Several tutorials were given:

- Tutorial on Timing Analysis at DATE 2005
- Tutorial on Timing Analysis and Timing Predictability at the ARTIST2 Summer school C

A chapter on Timing Analysis was authored by Reinhard Wilhelm in the Embedded Systems Handbook, CRC Press.

Partners in the cluster cooperate with research teams at Seoul National University, National University of Singapore, and Florida State University.

Tidorum and York (through Rapita Systems Ltd) are starting 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. This activity has been possible through the interaction started within the ARTIST2 network.