IST-214373 ArtistDesign
Network of Excellence
on Design for Embedded Systems

Activity - Progress Report for Year 1

Software Synthesis and Code Generation

Clusters:

SW Synthesis, Code Generation and Timing Analysis

Activity Leader:

Prof. Dr. Peter Marwedel (TU Dortmund)
http://ls12-www.cs.tu-dortmund.de/~marwedel

Policy Objective (abstract)

The objective of this activity is to provide software synthesis and code generation tools which are required for modern embedded architectures. Due to the constraints of such architectures, the tools have to generate very efficient code. A particular focus is on the mapping of applications to multi-processor systems on a chip (MPSoCs). The parallelism found in such architectures poses a particular challenge. In addition, other selected tools (linking, for example, timing analysis and compilation) are also considered.

Versions

number	comment	date
1.0	First version delivered to the reviewers	December 19th 2008

Table of Contents

1. Overview of the Activity 3
1.1 ArtistDesign Participants and Roles 3
1.2 Affiliated Participants and Roles 3
1.3 Starting Date, and Expected Ending Date 4
1.4 Policy Objective 4
1.5 Background 4
1.6 Technical Description: Joint Research 5
1.7 Problem Tackled in Year 1 7

2. Summary of Activity Progress 9
2.1 Technical Achievements 9
2.2 Individual Publications Resulting from these Achievements 11
2.3 Interaction and Building Excellence between Partners 12
2.4 Joint Publications Resulting from these Achievements 13
2.5 Keynotes, Workshops, Tutorials 14

3. Milestones, and Future Evolution 19
3.1 Problem to be tackled over the next 12 months (Jan 2009 – Dec 2009) 19
3.2 Current and Future Milestones 19
3.3 Main Funding 21

4. Internal Reviewers for this Deliverable 22

1. Overview of the Activity

1.1 ArtistDesign Participants and Roles

Prof. Dr. Peter Marwedel – TU Dortmund, Dortmund (Germany)
This team's role is to lead this activity, to work on resource aware compilation, worst-case execution time (WCET) aware compilation and to provide results on compilation for MPSoCs.

Dr. Stylianos Mamagkakis – IMEC, Leuven (Belgium)
This team will introduce novel source code parallelization and memory source-to-source optimizations for MPSoC platforms.

Prof. Dr. Christian Lengauer – U. Passau, Passau (Germany)
This team has advanced knowledge in program analysis techniques. This knowledge was initially generated in the context of high-performance computing and shall now be used for compilation to MPSoCs.

Prof. Dr. Rainer Leupers – RWTH Aachen, Aachen (Germany)
This partner's role is to work on compiler platforms, adaptive compilation, and MPSoC compilation. The group's MAPS project provides a reference for tools mapping algorithms to MPSoCs.

1.2 Affiliated Participants and Roles

Joseph van Vlijmen – ACE, Amsterdam (Netherlands)
This industrial partner is a key player in the compiler domain in Europe and the world. This partner provides a view on industrial requirements and practices.

Dr. Björn Franke – University of Edinburgh, Edinburgh (UK)
This team has advanced knowledge in program analysis techniques. This knowledge was initially generated in the context of high-performance computing and shall now be used, for example, for compilation to MPSoCs.

Prof. Dr. Sabine Glesner – TU Berlin, Berlin (Germany)
This team provides its expertise on program verification and compiler optimization to the network. This expertise will help verifying transformations as well as developing optimizing compiler transformations of single programs and applications.

Prof. Dr. Paul Kelly – Imperial College, London (UK)
This team has advanced knowledge in program analysis techniques. This knowledge was initially generated in the context of high-performance computing and shall now be used, for example, for compilation to MPSoCs.

Prof. Dr. Alain Darte – ENS, Lyon (France)
This team has advanced knowledge in program analysis techniques. This knowledge was initially generated in the context of high-performance computing and shall now be used, for example, for compilation to MPSoCs.

Dr. Marco Bekooji, Ruben van Royen – NXP, Eindhoven (Netherlands)
This partner's team has advanced knowledge in software synthesis from non-imperative models of computation. Within its area of expertise, the team has already designed tools mapping applications to multi-processors.

Dr. Bart Kienhuis – Compaan Design B.V., Leiden (Netherlands)
This partner's team has advanced knowledge in software synthesis from non-

imperative models of computation. Within its area of expertise, the team has already designed tools mapping applications to multi-processors.

1.3 Starting Date, and Expected Ending Date

This activity started with day 1 of the network. This activity includes the difficult problem of mapping applications to MPSoCs. We cannot expect that this problem will be completely solved at the end of the funding period. Therefore, work on this problem will be required for a number of years, even though the activity will formally be finished at the end of the funding period.

1.4 Policy Objective

Top-level experts have been selected for the activity. Their publication record, their reputation in industry and their links to leading colleagues clearly demonstrate that world class experts have been selected for the core teams. In order to achieve the required critical mass without increasing the number of partners beyond a manageable number, affiliated partners are added. These affiliated partners complement the work done by the core partners. For the same reason, external partners have been integrated into the work of the network. At some time, some external partners might become affiliated partners.

1.5 Background

Software synthesis and code generation tools are indispensable tools for developing embedded systems. They are frequently assumed to be available. New architectural features are introduced all the time, assuming that “somebody” will provide the expected tools. However, the design of such tools poses many very difficult challenges. There is always the risk of major losses of investments, if the expected tools cannot be designed in the available time.

Existing compilers represent very valuable software components which cannot be easily replaced by new methods. Many companies hesitate to replace their existing proven compilers by less well-debugged research results. Therefore, this cluster is extensively considering software synthesis and pre-pass source-to-source optimization tools, which can be used with several standard compilers. Pre-pass optimizers decouple the process of code generation and that of particular optimizations for certain architectural features.

The following is an enumeration of the background in the various areas of this activity:

1. Parallelism as available in MPSoCs is a particularly challenging new architectural feature. Significant effort on automatic parallelization has been spent in the context of high performance computing. Due to this effort, automatic parallelization has become feasible provided certain assumptions about the applications are met. The same results are not yet available for embedded systems. For embedded systems, the situation is different in various respects. MPSoCs, for example, are characterized by communication speeds which are comparable to the speeds of larger on-chip memories. As a result, communication based on the message-passing interface (MPI) is completely ill-designed, since it uses memory buffers extensively. Also, embedded system applications are different from general purpose or high performance computing. They are typically more “well-behaved” in that features like recursion, dynamic loop bounds, dynamic memory allocation, pointers, dynamic class loading etc. are much less frequent, simplifying the analysis. However, heterogeneity of processing elements, real-

time constraints, streaming data, limited communication resources and energy awareness impose additional restrictions.

2. Most embedded systems are integrated into a physical environment. In such an environment, time is frequently the most critical resource. It has been found that the lack of timing in the core abstraction of computer science is a serious flaw (see, for example, Ed Lee, ptolemy.eecs.berkeley.edu/publications/papers/05/APOT/APOT.pdf). Reconciling code generation and timing models should therefore receive more attention.
3. Efficiency of embedded systems is a main concern. Therefore, optimized architectures are required and resource allocation has to be handled with care. There are many proposals for optimizing architectures. For example, customized instruction sets, exploitation of attached FPGAs and multimedia instructions have been suggested. Most of these features require special consideration in compilers and code generation. Resource allocation includes the allocation of execution time, energy, memory space, bandwidth etc. Traditionally, these resources have been allocated independently. Integrated resource allocation is still not generally available. Memory allocation can be considered as a special case of resource allocation. Access times and energy consumption increase with the size of the memory. There is a growing gap between the speed of processors and the speeds of memories, even for larger on-chip memories of MPSoCs. Memory hierarchies are introduced to ease the problems resulting from this gap. Memory hierarchies are extremely important. Currently available memory hierarchies are typically designed to provide a good average-case performance. However, methods for increasing the average-case performance often deteriorate the worst-case performance and the timing predictability. Hence, timing predictability is becoming a key bottleneck for high-performance embedded systems and the memory system is a key source of unpredictability. Furthermore, memory hierarchies have not been designed for an efficient use of the available energy. In general, the link between memory architectures and compilation techniques is rather weak.
4. Software generation has evolved to a level where compilers are key components, but not the only components that are useful for generating executable code. New models of computations such as data-flow based models aim at avoiding the well-known disadvantages of imperative programming styles. Software synthesizers generate imperative code from abstract specifications such as Matlab, or Kahn process networks. It can also be expected that the link between software engineering and embedded systems will become stronger. Hence, trends like the use of UML-based system models do have to be respected as well. For the above models, code is synthesized from specifications in non-imperative languages.
5. Many applications in the embedded systems domain are not only resource-restricted but also safety-critical. This in turn requires compilers for embedded processors to be both efficient and correct. One crucial phase in the compiler is the code generation. Due to its complexity, further increased by reason of parallel processing which needs substantial support, it is highly error prone. Hence, verification of code generation is necessary to ensure that transformations preserve the semantics during compilation.

1.6 Technical Description: Joint Research

In order to make the results available to as many designers as possible, tools will be based on pre-pass source-to-source optimization tools whenever feasible. This way, the mapping of applications to MPSoCs can be added to many existing, proven tool flows. Investments into compilers can be protected, the development effort can be reduced and the focus on new optimization techniques can be increased. The key advantage of pre-pass optimizers is their

applicability in a large number of tool chains. Such tool chains do not require new compilers to be written. They may be using a compiler from a family of compilers (such as gcc) or specially designed compilers. Pre-pass optimizers can easily support a family of compilers without any modification and different compilers with only few modifications. Pre-pass optimizers do already exist for memory-architecture aware compilation and program parallelization. IMEC and partners at the Universities of Dortmund, Passau, and Edinburgh have significant experience with the design of pre-pass optimizers. They reflect the fact that the resources of a network of excellence are limited.

The following joint work with a focus on integration has been or will be performed in this activity:

1. *Compilation techniques for MPSoCs* cannot be developed from scratch since the problems to be solved are very challenging. The current project will certainly not provide enough resources to develop completely new techniques. Fortunately, we can build upon compilation techniques for high-performance computing. Using the limited resources, we established a link between the high-performance computing and the embedded system domain. Integration activities will comprise an in-depth analysis of the applicability of techniques designed in one domain to the other domain. For this purpose, it is very essential that the proposed project includes enough expertise in different areas of applications. Knowledge about hardware architectures would not be sufficient to really check the applicability of the techniques. The University of Passau is a link to the high-performance community. Cooperation with core partners and selected affiliates is used to check which of the existing techniques can be employed in embedded systems and which extensions are needed. Twelve months after the start of the network, a plan for integration of tools and for closing gaps was expected. After 24 months, the design of the integration work should be complete. Later, available automatic parallelization techniques will be integrated to close identified gaps in the tool support. After 36 months, the implementation of the integrated tools should be complete. After 48 months, an evaluation of the integration should be available. This area has not been tackled in the Artist2 network of excellence.
2. Reconciliation of compilers and timing analysis bridges the two activities of this cluster. The work in this area builds on top of the integration work performed in the Artist2 network of excellence. The existing integration of timing analysis and compilers will be used to explore the potential of this approach further. Additional information can be passed between compilers and timing analysis. The impact of optimizing for WCET has to be studied further. Additional hardware components have to be studied. The influence of context switches has to be analysed. Techniques for reducing the number of calls of timing analysers would be of interest.
3. Design efficiency will be dealt with in ArtistDesign. Design efficiency can be achieved with many different means. Research on specialized instructions and new optimizations can be expected. If feasible, such optimizations will be implemented as pre-pass optimizations so that they can be used with various compilers. Memory architectures will be considered in-depth, due to their potential for contributing toward an overall efficiency. Memory architectures will also be very important for the mapping to networked processors. Indeed, the mapping of applications to processing elements may be significantly affected by the connectivity of the memories. Hence, optimized mappings to memories have to be considered as well. Such techniques should provide optimization techniques taking several objectives into account. Work in this area builds on top of previous work in the Artist2 network.
4. Software synthesis will be a long-term goal. Software synthesis has not been considered in the compiler cluster of Artist2. As much as feasible, it will be considered as well in ArtistDesign. The activity includes affiliate partners specializing on software

synthesis. Also, other clusters of the network include prominent experts on software synthesis (Benveniste, Halbwachs). The potential resulting from this will be explored. However, the activity includes a very limited set of partners. Thus, we have to set priorities regarding the research areas to be covered and, unfortunately, software synthesis cannot receive a high priority.

5. In addition to topics 1 to 4, members of this activity will also extend the areas for which the correctness of compilers has been shown via formal verification, focussing on the crucial code generation phase during compilation. The work in this area builds on top of the work performed in the Artist2 network of excellence.
6. The members of this activity will also contribute to the thematic activities of the Transversal Integration work package, focussing on predictability and adaptivity issues. This area has not been tackled in the Artist2 network.

1.7 Problem Tackled in Year 1

The following work was performed for the different areas of the activity:

1. It was important to set up the required interaction of the partners regarding compilation for MPSoCs. The partners started with an intensive workshop in June, 2008. The workshop was held from June 16 to June 17, 2008 at Rheinfels Castle, St. Goar, Germany. Due to the complexity of the problem and the limited manpower of the network, we invited a number of European groups known to be working in relevant areas to the workshop. All members of the HIPEAC network of excellence were invited. The same applies for members of the ACOSTES project. This way, we tried to reach out far beyond the limited set of ArtistDesign partners, using these partners as the seed for a larger network of cooperating partners. A total of 37 attendees (representing 24 different groups) were present at the workshop, including industrial attendees e.g. from ARM and NXP. The workshop was organized as a closed, interactive workshop. The workshop provided a very good overview over the work that was performed so far. Presentations included approaches starting from sequential von Neumann languages as well as presentations using non-standard models of execution. Slides from the workshop are available at the ArtistDesign website (see http://www.artist-embedded.org/artist/Mapping-of-Applications-to-MPSoCs.html). Two classes of existing work were identified: design space exploration and code generation. Members of the network were associated with these classes. The workshop was meant to start a series of internationally visible workshops. This goal was achieved: the cluster leader gave an invited presentation summarizing the main contributions at the CASA workshop held as part of the Embedded Systems Week in Atlanta on Oct. 19th, 2008. The presentation was well received.

   The attendees found that a working meeting would be required in order to fully explore the potential for cooperation. The working meeting took place at Düsseldorf from Nov. 27 to Nov. 28, 2008. In-depth demonstrations of available tools were used to identify more precisely the joint potential for solving the mapping problems. Two additional groups were represented at the workshop. The partners agreed to intensify their cooperation. Several mini-clusters were identified, which will be combining their expertise.

   Also, the attendees of the 1st workshop agreed to have an open follow-up workshop at Rheinfels Castle in June 2009.

   In addition, several smaller meetings took place. For example, Olivera Jovanovic (TU Dortmund) visited the group of Lothar Thiele at ETHZ on Nov. 14th, 2008. The goal of this meeting was to explore opportunities for extending the DOL tools from Zürich by a

more detailed analysis of memory architectures. At the Rheinfels workshop, the DOL tools were identified as a possible starting point for mapping tools. However, DOL does not support a detailed analysis of the memory architecture. The memory architecture has an important influence on how the mapping should be done, but a full incorporation of the memory architecture could potentially lead to an explosion of the times required for architecture optimization. Good compromises have to be found.

This work was performed in cooperation with the execution platforms cluster and involved teams outside the ArtistDesign network.

2. Regarding the reconciliation of compilers and timing analysis, significant work was performed as well. The key questions to be solved were: how much would a compiler benefit from a tight integration with timing analysis? How much different are the code generation results for an optimization of the average case and of the worst case? Will the average run time increase if we optimize for the worst case?
3. Regarding the work on design efficiency, the problem tackled concerned the integration of various tools from various partners (not just limited to this activity).
4. For software synthesis, the question was how to find a link to the work on mapping of applications to MPSoCs.
5. Concerning the support for verification of code generation, TU Berlin continues its work on formalizing important parts of the semantics of the intermediate representation within the compiler and of the assembler code. As a result, the code generation could be proven as correct for a subset of the language.
6. For transversal integration, getting requirements from industry was a key goal

Resources of the network were used to support the cooperation. Research work was paid through other resources (see section 3.3).

2. Summary of Activity Progress

2.1 Technical Achievements

Definition of a mapping tool chain (TU Dortmund, ICD Dortmund, IMEC vzw, TU/e, DUTH)
In cooperation with the Mnemee project through the 7th framework, partners defined a tool chain starting with either sequential algorithms or task graphs. This tool chain shall comprise tools for mapping applications to MPSoCs and tools for optimizing the mapping to memories. The plan is to have independently usable tools which can also be tightly integrated. The tool chain shall incorporate a tool for automatic parallelization. For this, tools presented at the Rheinfels workshop serve as a reference. Existing tools from partners will be integrated where appropriate. Mnemee partners focus on the memory hierarchy, while ArtistDesign partners focus on the mapping to processors. The design of these tools will be the next step. This achievement concerns item 3 of sections 1.5 to 1.7.
http://www.mnemee.org

Memory hierarchy allocation optimization (IMEC vzw, DUTH)
In cooperation with the Mnemee project through the 7th framework, partners defined a source-to-source tool chain that will connect static and dynamic memory hierarchy allocation optimization for MPSoCs. More specifically, the MATISSE framework for dynamic data type and dynamic memory management optimizations will be linked with the MH tool for static Block Transfer and static memory assignment optimization. This achievement concerns item 3 of sections 1.5 to 1.7.

Source code parallelization for MPSoC platforms with NoC interconnect (IMEC vzw, DUTH, KTH)
In cooperation with the Mosart project through the 7th framework, partners defined a tool chain that will connect automatic parallelization tools and data access optimization tools for MPSoC with Network-on-Chip interconnect. More specifically, the MATISSE framework for dynamic data type optimizations will be extended and linked with the Nostrum NoC optimization methodology. Both of them will be linked with the MPA tool for automatic source code parallelization. This achievement concerns item 3 of sections 1.5 to 1.7.

System scenarios for embedded system design (IMEC vzw, TU/e)
This particular collaboration is relevant for both the SW Synthesis, Code Generation and Timing Analysis cluster and the Hardware Platforms and MPSoC cluster. More details can be found on the Hardware Platforms and MPSoC Design activity deliverable of Year 1.

Integration of timing analysis and compilation (TU Dortmund, AbsInt)
The integration of the aiT timing analysis tool from AbsInt into the experimental worst-case execution time (WCET) aware compiler WCC has been continued. The resulting compiler can be considered the leading WCET-aware compiler. The integrated tool set allowed studying the impact of optimizations for WCET minimization at high and at low levels of abstraction of the code. This achievement concerns item 2 of sections 1.5 to 1.7.
http://ls12-www.cs.tu-dortmund.de/publications

Extension of MPARM (TU Dortmund, Bologna)
MPArm is a simulator designed at Bologna. TU Dortmund extended this simulator to allow the modelling of more complex memory architectures. The new version is now stored in a repository at Bologna. ETH Zürich intends to also use this simulator. This achievement concerns item 3 of sections 1.5 to 1.7.

Software synthesis for MPSoCs (NXP, Compaan, RWTH Aachen)
Partners from NXP and Compaan participated in the 1st workshop on mapping applications to MPSoCs. Some potential links between software synthesis and MPSoCs became clear. Partners from NXP visited RWTH Aachen. This achievement concerns item 4 of sections 1.5 to 1.7.
http://www.artist-embedded.org/artist/Mapping-of-Applications-to-MPSoCs.html

Development and verification of compiler transformations (TU Berlin, ACE, University of Edinburgh)
In cooperation between the TU Berlin and ACE, verification methods and tools for compilers have been investigated. In particular, TU Berlin developed new compiler optimization techniques using supervised classification learning to overcome the impact of the memory wall. To this end, TU Berlin continued their cooperation with Edinburgh University (Björn Franke). For supervised classification, it is necessary to select an appropriate training set for learning. In case of machine learning used for compiler optimizations, training sets are programs to be compiled. The approach taken by the TU Berlin is to group similar programs into program classes to obtain applicable training sets. Furthermore, important parts of the semantics of the intermediate language and the assembler code were formalized in order to prove the code generation as correct. This achievement concerns item 5 of sections 1.5 to 1.7.

Setting up the transversal activities (TU Dortmund, partners of the Transversal Cluster)
The cluster leader participated in the meeting organized by the leaders of the transversal activity regarding industrial requirements in Rome on Nov. 12th and 13th, 2008. Please refer to the reports by the transversal activity cluster for achievements. Work on timing predictability continued at TU Dortmund. This achievement concerns item 5 of section 1.7.

Making polyhedral loop parallelization go multicore (U. Passau)
Several steps have been taken to make the polyhedron model for loop parallelization fit the requirements of code generation for multi-cores better. The general aim is to break the barriers of affinity in the mathematical foundation of the model. Affinity is required in the loop bounds an in the defining equations of the parallelizing transformation. In both case, affinity could be transcended. This work is expected to find applications by partners aiming at the generation of parallel code.

Laying a foundation for feature-oriented software design (U. Passau)
Feature orientation is a new paradigm which supports the largely automatic generation of software systems. A feature is a unit of functionality which may affect many modules in a software system with modular structure. Adding a new feature to a software system by hand is difficult and error-prone. The DFG-funded project FeatureFoundation is laying the ground work for an algebraic approach to feature orientation. The concept of a feature fits well with the different demands of functionality in an embedded system.

2.2 Individual Publications Resulting from these Achievements

RWTH Aachen

- Ceng, J., Castrillon, J., Sheng, W., Scharwächter, H., Leupers, R., Ascheid, G. and H. Meyr, RWTH Aachen University, Isshiki, T. and H. Kunieda, Tokyo Institute of Technology. "*MAPS: An Integrated Framework for MPSoC Application Parallelization*". In 45th Design Automation Conference (DAC '08), Anaheim, CA, USA, June 2008
- Anupam Chattopadhyay, Heinrich Meyr, Rainer Leupers: *LISA: A Uniform ADL for Embedded Processor Modelling, Implementation, and Software Tool suite Generation*, in Prabhat Mishra, Nikil Dutt (ed.): *Processor Description Languages - Applications and Methodologies*, Morgan Kaufmann, 2008

TU Dortmund

- Paul Lokuciejewski, Heiko Falk, Peter Marwedel: *WCET-driven Cache-based Procedure Positioning Optimizations*, Proceedings of the 20th Euromicro Conference on Real-Time Systems (ECRTS), Prague, Czech Republic, July, 2008.
- Peter Marwedel: *MIMOLA - A fully synthesizable language*, in: Prabhat Mishra, Nikil Dutt (ed.): *Processor Description Languages - Applications and Methodologies*, Morgan Kaufmann, 2008
- Heiko Falk (Editor), *Proceedings of the 11th International Workshop on Software & Compilers for Embedded Systems (SCOPES)*, Munich, Germany, March 2008.
- Paul Lokuciejewski, Fatih Gedikli, and Peter Marwedel: *Accelerating WCET-driven optimizations by the Cold Path Paradigm – a Case Study of Loop Unswitching*, Submitted to SCOPES, 2009.
- Paul Lokuciejewski, Daniel Cordes, Heiko Falk, Peter Marwedel: *A Fast and Precise Static Loop Analysis based on Abstract Interpretation, Program Slicing and Polytope Models*, International Symposium on Code Generation and Optimization (CGO), 2009.
- Sascha Plazar, Paul Lokuciejewski, Peter Marwedel: *A Retargetable Framework for Multi-objective WCET-aware High-level Compiler Optimizations*, IEEE Real-Time Systems Symposium (RTSS/WIP), 2009.
- Paul Lokuciejewski, Fatih Gedikli, Peter Marwedel, Katharina Morik: *Automatic WCET Reduction by Machine Learning Based Heuristics for Function Inlining,* Submitted to 3rd HIPEAC Workshop on Statistical and Machine Learning approaches applied to Architectures and Compilation (SMART), 2009

IMEC

- M. Palkovic, H. Corporaal, F. Catthoor: *Dealing with data dependent conditions to enable general source code transformations*, International Journal of Embedded Systems, 2008
- P. Kjeldsberg, F. Catthoor, S. Verdoolaege, M. Palkovic, A. Vandecappelle, Q. Hu, E. Aas: *Guidance of loop ordering for reduced memory usage in signal processing Applications*, Journal of VLSI Signal Processing Systems, 2008
- F. Balasa, P. Kjeldsberg, A. Vandecappelle, M. Palkovic, Q. Hu, H. Zhu, F. Catthoor: *Storage estimation and design space exploration methodologies for the memory management of signal processing applications*, Journal of VLSI Signal Processing Systems, 2008
- I. Issenin, E. Brockmeyer, B. Durinck, N. Dutt: *Data-reuse driven energy-aware cosynthesis of scratch pad memory and hierarchical bus-based communication*

architecture for multiprocessor streaming applications, IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems, 2008

- R. Baert, E. De Greef, E. Brockmeyer, G. Vanmeerbeeck, P. Avasare, J. Mignolet, M. Cupak: *An automatic scratch pad memory management tool and MPEG-4 encoder case study*, 45th Design Automation Conference (DAC), 2008

U. Passau

- A. Größlinger: *Some Experiments on Tiling Loop Programs for Shared-Memory Multicore Architectures*, in A. Cohen, M. J. Garzaran, C. Lengauer, S. Midkiff (eds.) *Programming Models for Ubiquitous Parallelism*, Dagstuhl seminar Proceedings 07361, Schloss Dagstuhl, Germany, 2008.
- A. Größlinger: *Scanning Index Sets with Polynomial Bounds Using Cylindrical Algebraic Decomposition*, Technical Report MIP-0803, University of Passau, June 2008.
- A. Größlinger, S. Schuster: *On Computing Solutions of Linear Diophantine Equations with One Non-Linear Parameter*, in Proceedings of the 10th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC 2008), IEEE Computer Society Press, to appear.
- A. Größlinger: *Precise Management of Scratchpad Memories for Localising Array Accesses in Scientific Codes*, submitted to Compiler Construction (CC 2009).
- S. Apel, C. Kästner, C. Lengauer:. *Feature Featherweight Java: A Calculus for Feature-Oriented Programming and Stepwise Refinement*. Proceedings of the ACM International Conference on Generative Programming and Component Engineering (GPCE), ACM Press, October 2008..
- S. Apel, C. Lengauer, B. Möller, C. Kästner: *An Algebra for Features and Feature Composition*, Proceedings of the International Conference on Algebraic Methodology and Software Technology (AMAST), volume 5140 of Lecture Notes in Computer Science, pages 36–50. Springer-Verlag, July 2008.
- S. Apel, C. Lengauer: *Superimposition: A Language-Independent Approach to Software Composition*. In Proceedings of the ETAPS International Symposium on Software Composition (SC), volume 4954 of Lecture Notes in Computer Science, pages 20–35. Springer-Verlag, March 2008.

TU Berlin

- L. Gesellensetter, S. Glesner, E. Salecker: *Formal Verification with Isabelle/HOL in Practice: Finding a Bug in the GCC Scheduler*, Formal Methods for Industrial Critical Systems (FMICS'07), Revised Selected Papers, Springer LNCS 4916, 2008.
- L. Gesellensetter, S. Glesner: *Interprocedural Speculative Optimization of Memory Accesses to Global Variables*, Euro-Par 2008, Springer LNCS 5168, 2008.

2.3 Interaction and Building Excellence between Partners

Main interaction between the partners was through the 1st workshop on the mapping of applications to MPSoCs on June 16-17, 2008 and the follow-up working meeting at Düsseldorf on Nov. 27-28, 2008.

TU Dortmund is cooperating with ETH Zürich on exploring the idea of extending the design space exploration from ETZ Zürich with memory-aware techniques. For this reason, researcher Olivera Jovanovic visited ETH Zürich in mid-November.

The partners from Dortmund (at ICD), Leuven (at IMEC) and Eindhoven (at TU Eindhoven, member in another ArtistDesign cluster) are jointly working on the MNEMEE project funded through the 7th framework (see http://www.mnemee.org). TU Eindhoven has integrated the compiler development framework ICD-C (see http://www.icd.de/es/index.html) into some of its tools.

IMEC is cooperating with TU/e and DUTH (affiliated partners in another ArtistDesign cluster) on system scenarios for MPSoC system level design optimizations (see http://www.es.ele.tue.nl/~vali/scenarios/).

The partners from IMEC, DUTH and KTH are jointly working on the MOSART project funded through the 7th framework (see http://www.mosart-project.org).

The partners from RWTH Aachen and TU Dortmund are jointly teaching a course in retargetable compilation (including memory-architecture aware compilation) at the Advanced Learning and Research Institute (ALARI) in Lugano, Switzerland (see http://www.alari.ch).

The partners from TU Dortmund and ETH Zürich (member in another ArtistDesign cluster) have jointly taught at the South American Summer School on Embedded Systems (see http://www.artist-embedded.org/artist/Objectives,1365.html).

The partners from Dortmund and members of other ArtistDesign activities (Bologna, Pisa, Saarbrücken, Zürich) are jointly working on the PREDATOR project funded through the 7th framework (see http://www.predator-project.eu).

RWTH Aachen and ACE are co-operating on loop parallelization techniques for embedded processors. Max Ferger, Diploma-student from RWTH Aachen, did an internship from June to Nov.. 2008 working on this topic in the CoSy compiler framework of ACE.

The partners from RWTH Aachen are also members of the HIPEAC Network of Excellence where they lead the research cluster on Design Methodology and Tools. Within the HIPEAC NoE and carrying the ArtistDesign banner they have interacted with top level academic and industrial partners. On May 15th, 2008, a one-day meeting was held at the University of Edinburgh with partners Prof. Nigel Topham and Dr. Björn Franke. During the meeting, discussions on MPSoC programming methods and design of customized embedded processors took place. On July 1st, 2008, partners from THALES, Eric Lenormand and Sami Yehia, visited the RWTH Aachen partners and discussed cooperation possibilities on MPSoC programming tools and models of computation for MPSoCs. On August 26th 2008, partners from NXP Eindhoven led by Marco Bekooij visited the RWTH Aachen partners and discussed topics related to Real Time Scheduling for MPSoCs and the problem of multi-application mapping and scheduling under timing constraints.

TU Berlin and ACE extended their cooperation concerning the development of optimizing compiler transformations as well as verifying transformations. Besides, TU Berlin visited and was visited by other cluster members, e.g. ACE, University of Edinburgh, and RWTH Aachen.

2.4 Joint Publications Resulting from these Achievements

Aachen, ACE

- M. Hohenauer, F. Engel, R. Leupers, G. Ascheid, H. Meyr, RWTH Aachen University; G. Bette, ACE; B. Singh, NXP Semiconductors Eindhoven: *Retargetable Code Optimization for Predicated Execution*, DATE, Munich, Germany, March 2008.

TU Dortmund / AbsInt

- Paul Lokuciejewski, Heiko Falk, Peter Marwedel, Henrik Theiling: *WCET-Driven, Code-Size Critical Procedure Cloning*, Proceedings of the 11th International Workshop on

Software & Compilers for Embedded Systems (SCOPES), Munich, Germany, March, 2008.

- Niklas Holsti, Jan Gustafsson, Guillem Bernat (eds.), Clément Ballabriga, Armelle Bonenfant, Roman Bourgade, Hugues Cassé, Daniel Cordes, Albrecht Kadlec, Raimund Kirner, Jens Knoop, Paul Lokuciejewski, Nicholas Merriam, Marianne de Michiel, Adrian Prantl, Bernhard Rieder, Christine Rochange, Pascal Sainrat, Markus Schordan, *WCET TOOL CHALLENGE 2008: REPORT*, Proceedings of the 8th International Workshop on Worst Case Execution Time Analysis (WCET), 2008

IMEC vzw., TU/e

- S.V. Gheorghita, M. Palkovic, J. Hamers, A. Vandecappelle, S. Mamagkakis, T. Basten, L. Eeckhout, H. Corporaal, F. Catthoor, F. Vandeputte, K. De Bosschere: *A System Scenario based Approach to Dynamic Embedded Systems*, ACM Transactions on Design Automation of Electronic Systems, ToDAES. To appear in 2009.

IMEC vzw., TU/e, DUTH, TU Dortmund (at ICD)

- S. Mamagkakis, P. Lemmens, D. Soudris, T. Basten, P. Marwedel, D. Kritharidis, G. Guilmin: *MNEMEE: Memory management technology for adaptive and efficient design of embedded systems*, 16th IFIP/IEEE International Conference on Very Large Scale Integration - VLSI-SOC, 2008.

IMEC vzw., KTH, DUTH

- B. Candaele, A. Jantsch, T. Ashby, K. Tiensyrjä, F. Ieromnimon, B. Vanthournout, P. Di Crescenzo, D. Soudris: *MOSART: Mapping Optimisation for Scalable multi-core ARchiTecture*, 16th IFIP/IEEE International Conference on Very Large Scale Integration - VLSI-SOC, 2008.

2.5 Keynotes, Workshops, Tutorials

Course: Peter Marwedel, Rainer Leupers: Retargetable Compilation

Lugano, Switzerland, Feb. 25-29., 2008

The course consisted of two parts: the first part (by Peter Marwedel) focused on memory-architecture aware compilation. The second part (by Rainer Leupers, RWTH Aachen) focused on processor retargetability. The course was supported by ALARI.

http://www.alari.ch

Tutorial: Peter Marwedel, Embedded Systems in a Nutshell, Spring School on Knowledge Discovery in Ubiquitous Systems,
Porto, Portugal, March 2, 2008

This tutorial provided a brief overview over specification techniques, hardware, scheduling and optimization of embedded systems for a community without any pre-existing knowledge on embedded systems.

http://www.kdubiq.org

Tutorial: Rainer Leupers, Gerd Ascheid (RWTH Aachen), Wilfried Verachtert, Tom Ashby, Arnout Vandecappelle (IMEC): System-Level Design and Application Mapping for Wireless and Multimedia MPSoC Architectures
DATE 2008
Munich, Germany, March 10, 2008

Advanced embedded devices such as multi-standard mobile terminals demand ever-increasing performance and energy efficiency. Simultaneously, a high degree of flexibility and programmability is required due to increasing software complexity and fast changing protocol

and codec standards. This has led to the concept of MPSoC (Multi-Processor System-on-Chip) platforms. In many cases, MPSoCs are simply assembled in "best effort" manner from existing legacy IP components, and programming the platform presents a major bottleneck. As Moore´s Law permits us to enter the "many core" MPSoC area, what is needed is a systematic approach that builds on well-proven technologies, but also innovates with novel classes of electronic system-level (ESL) design automation tools.

This tutorial discussed several key questions with significant impact on the future of MPSoC: What are the MPSoC killer applications? Is homogeneous or heterogeneous architecture the right choice? What are the key tools, methodologies and programming models for successfully designing and programming MPSoC platforms? In the end, will there be only a few survivor platforms that everyone has to accept? Based on their extensive research and industry experience, the presenters provided their answers from a practical, application-oriented perspective.

http://www.date-conference.com/archive/conference/proceedings/PAPERS/2008/DATE08/PDFFILES/TUTORIALS.PDF

Workshop: Software & Compilers for Embedded Systems (SCOPES) 2008
Munich, Germany – March 13-14, 2008

SCOPES focuses on the software generation process for modern embedded systems. Topics of interest include all aspects of the compilation process, starting with suitable modelling and specification techniques and programming languages for embedded systems. The emphasis of the workshop lies on code generation techniques for embedded processors. The exploitation of specialized instruction set characteristics is as important as the development of new optimizations for embedded application domains. Cost criteria for the entire code generation and optimization process include run time, timing predictability, energy dissipation, code size and others. Since today's embedded devices frequently consist of a multi-processor system-on-chip, the scope of this workshop is not limited to single-processor systems but particularly covers compilation techniques for MPSoC architectures.

In addition, this workshop puts a spotlight on the interactions between compilers and other components in the embedded system design process. This includes compiler support for e.g. architecture exploration during HW/SW codesign or interactions between operating systems and compilation techniques. Finally, techniques for compiler aided profiling, measurement, debugging and validation of embedded software were also covered by this workshop, because stability of embedded software is mandatory.

SCOPES 2008 was the 11th workshop in a series of workshops initially called "International Workshop on Code Generation for Embedded Processors". The name SCOPES has been used since the 4th workshop. The scope of the workshop remains software for embedded systems with emphasis on code generation (compilers) for embedded processors.

SCOPES 2008 was organized by Heiko Falk from TU Dortmund and was held as DATE Friday Workshop. There were many discussions between cluster members at SCOPES (starting already on the eve before the sessions), at DATE and during an Artist2 meeting during the same week, making the entire week the key joint event in spring.

http://www.scopesconf.org/scopes-08

Keynote: Rainer Leupers: ESL Design Technologies for Wireless and Multimedia MPSoC Architectures
3rd International Symposium on Industrial Embedded Systems (SIES 2008)
La Grande Motte, June 11-13, 2008
http://www.lirmm.fr/SIES2008/

Meeting: 1st Workshop on Mapping Applications to MPSoCs, 2008
St. Goar*, Germany – June 16-17, 2008*

Objectives for the meeting: The goal of the ArtistDesign workshop was to identify requirements and partial solutions for the problem of mapping applications to MPSoCs. It was considered to be the starting point for more intensive cooperations in the ArtistDesign framework. Also, members of other projects (HIPEAC2, ACOTES) were invited.
Organizer: Peter Marwedel (TU Dortmund)
Other participants: there were a total of 37 participants.
Conclusions: The topic was partitioned into two related areas: mapping and code generation. Working groups were formed and it was agreed to have joint follow-up workshops.

http://www.artist-embedded.org/artist/Mapping-of-Applications-to-MPSoCs.html

Meeting: Working Meeting IFIP 2.11 (Program Generation)
Passau, *Germany – Jun. 19-21, 2008*
Objectives for the meeting: General discussion on the future of program generation.
Organizer: Christian Lengauer (U. Passau)
Other participants: IFIP 2.11.

Forum: 8th International Forum on Application-Specific Multi-Processor SoC, 2008
Aachen, Germany, 23 - 27 June 2008,

MPSoC is a pluridisciplinary forum bringing together key R&D actors from the different fields required to design heterogeneous multiprocessor SoC (MPSoC). MPSoC '08, 8th event of the forum series, was held on 23-27 June 2008 at Château St. Gerlach (near Aachen) and was organized by ISS/SSS, RWTH Aachen University. Rainer Leupers from RWTH Aachen was one of the general co-chairs of this premier event. The full week format and the quality of both attendees and speakers made MPSoC '08 a unique occasion for executives and senior managers to explore new ideas and refine strategic thinking. The program brought together key actors from IP, fabless, semiconductor, system houses and design industry to build a vision of the next step in integrated system design. More than 50 world class R&D speakers discussed fundamental and strategic issues to master multi-processor SoC design. This year's technical sessions presented strategic directions and state-of-the-art research, covering topics like MPSoC Architecture, MPSoC Application Platforms, MPSoC Programming, MPSoC Design methodologies, etc. The detailed program and the slides from the speakers can be found in the event's website.

Organizer: Rainer Leupers, Heinrich Meyr (RWTH Aachen)

http://www.mpsoc-forum.org

Meeting: Working Meeting on Timing Analysis.
Prague*, Czech Republic – July 2nd, 2008*
Objectives for the meeting: to follow up on the timing analysis discussions held at the Rheinfels workshop, and to discuss ideas for future collaborative research.
Organizer: Björn Lisper (Mälardalen U.)
Other participants: this was an internal ARTIST meeting held in conjunction with the WCET Workshop.
Conclusions: A number of ideas for potential research collaborations came up. It was decided to try to set up a common meeting with the HW Platform and MPSoC Design cluster.

Invited Course: Peter Marwedel, Heiko Falk, Embedded Systems with Emphasis on the Exploitation of the Memory Hierarchy
Advanced Institute of Information Technology
Seoul, Korea – August 11-15, 2008

The goal of this course is to provide an overview over key areas in embedded system design which should be taught at Universities. After attending the course, the attendees should be able to compare different approaches to embedded system design education and their

advantages and limitations. The attendees will also become familiar with the contents of a course on embedded system design which aims targets second or third year students. The course should enable attendees to design the structure of embedded system education at their Universities. In the last third of the course, attendees will be introduced to research topics regarding embedded system optimization. In particular, this last third will address the so-called memory wall problem (the problem resulting from the small performance improvements of memories). This problem is frequently seen as the key problem for further performance enhancements of future systems. This material would be appropriate for an advanced course in embedded system design.

Peter Marwedel and Heiko Falk from TU Dortmund lectured this one-week course for Korean professors (CS and EE) after an invitation by the Korean Advanced Institute of Information Technology.

http://ttt.aiit.or.kr

Tutorial: Peter Marwedel: Memory architecture aware compilation for Embedded Systems
Artist South American Summer School
Florianopolis, Brazil, Aug. 25.-29., 2008

The tutorial focused on compilation techniques exploiting descriptions of the memory architecture.

http://www.artist-embedded.org/artist/Objectives,1365.html

Invited Talk: Peter Marwedel, Heiko Falk: Memory architecture aware compilation
Autrans, France, Sept. 10, 2008-11-20

This talk gave an overview over compilation for scratchpad memories and linked it to worst case execution time aware compilation.

Invited talk: Peter Marwedel: Mapping of Applications to MPSoCs
4th Compiler Assisted SoC Assembly Workshop (CASA08)
Atlanta, USA – Oct. 19th, 2008

The talk summarized the presentations of the "1st workshop on the mapping of applications to MPSoCs" for a wider audience.

http://www.esweek.org/

Workshop: 4th Workshop on Embedded Systems Education, 2008
Atlanta, US, – October 23, 2008

Embedded system education is still a very young area and frequently restricted to teaching the details of microcontroller programming. A long-term objective of this workshop is to stimulate the introduction of broader curricula.

http://www.esweek.org/

Meeting: Working Meeting on Mapping Applications to MPSoCs (*Fall Activity meeting*).
Düsseldorf, *Germany – Nov. 27-28, 2008*
Objectives for the meeting: The goal of the ArtistDesign meeting was to intensify the discussions started at the Rheinfels workshop and to advance the cooperation between the partners.
Organizer: Peter Marwedel (TU Dortmund)
Other participants: there were a total of 21 participants.
Conclusions: Details of the cooperation between the partners were fixed.

Keynote: Rainer Leupers: Advanced MPSoC design technologies in the UMIC project

J-CING (Japan – CoWare Innovators Group) 2007
Tokyo, Nov. 30, 2007
http://www.coware.co.jp/J-CING/keynote2.html

3. Milestones, and Future Evolution

3.1 Problem to be tackled over the next 12 months (Jan 2009 – Dec 2009)

1. Consistent with the description of the work (DoW), a potential road toward the mapping of applications to MPSoCs has been identified in the first year. This road will now have to be build. The key goal is to really come up with the identified tools. Since ArtistDesign mainly supports integration, they will have to be based on integrated (and possibly extended) existing tools or the tools have to be designed through other resources for the required manpower. Integration is foreseen for tools from ETH Zürich and TU Dortmund.
2. The impact of WCET minimization will be analyzed for additional optimizations. The impact of the memory architecture on the WCET will be studied.
3. The MPARM simulator, initially designed at Bologna, will be used by a number of partners. The integration of memory-architecture-aware compilation tools will be completed.
4. Software synthesis will be dealt with at the Rheinfels workshop in 2009.
5. Concerning the improvement of results from statistical learning to guide compiler optimizations, program classification will be considered to select training sets. Besides, further parts necessary for verifying the code transformation during compilation will be formalized in a theorem prover.
6. The partners will also contribute to other activities, such as the transversal activities.

3.2 Current and Future Milestones

The following list of current and future milestones used the same enumeration which was used in sections 1.5 to 1.7.

1. **Mapping of applications to MPSoCs**
    a. The identification of the needs and possible approaches was the focus of the first year. The goal was to come up with ideas for the design of mapping tools and to see how existing tools could be integrated into a flow meeting the requirements. *This goal has been achieved. The results from the Rheinfels workshop ((*see http://www.artist-embedded.org/artist/Mapping-of-Applications-to-MPSoCs.html) *together with the results from the working meeting at Düsseldorf provide sufficient input (see* http://www.artist-embedded.org/artist/map2mpsoc-08-fall-meeting.html).
    b. The **design of an integrated tool flow** will be the goal for the second year. This design will comprise the key decisions regarding interfaces and implementations. It will not include a full implementation.
    c. The implementation of an integrated tool flow will be the goal for the third year. This implementation will be based on the design to be available at the end of the second year.
    d. The evaluation of an integrated tool flow will be the goal for the fourth year.

2. **Reconciliation of timing analysis and compilers**
    a. The impact of WCET-aware compilation on the resulting code should be analyzed. *This goal has been achieved. Several WCET-aware optimizations have been designed and their impact on the code quality has been published.*
    b. The **analysis of the impact of WCET-aware compilation** on the resulting code should be continued in the second year, since many of the standard optimizations have not been considered yet.
    c. Approaches for reducing the time required for WCET analysis during compilation should be proposed.
    d. A critical evaluation of advantages and limitations of WCET-aware compilation should be performed.
3. **Resource-aware compilation**
    a. Support for predicated execution should be available. Support for extended modelling of memory architectures in the MPARM simulator should be available. An interface for driving simulators and optimizers from the same architectural description should be designed. *These goals have been achieved. A paper on exploiting predicated execution has been published by RWTH Aachen. The extended version of MPARM has been submitted to Bologna by TU Dortmund. The MACC framework for architecture-aware compilation has been designed.*
    b. MPARM should be used as a key simulation tool by an extended set of partners. Additional support for special features of embedded processors should be available. Areas 2 and 3 should be linked by **exploiting memory hierarchies in WCET-aware compilers**. This work should be performed in year 2 of the network.
    c. Results from the integration in year 2 should be available in year 3.
    d. At least one source-to-source transformation tool exploiting memory hierarchies should be made publicly available in year 4.
4. **Software synthesis**
    a. The impact of software synthesis should be analysed. *This goal has been only partially achieved. NXP and Compaan presented their results at the Rheinfels workshop. However, due to the limited number of partners in this cluster, many of the specialists in this area could not be involved.*
    b. Due to the increasing importance of non-standard models of computation, **software synthesis** will be continued as a sidetrack.
5. **Development and Verification of compiler transformations**
    a. The goal here is to facilitate compilers to generate efficient and correct code. *This goal has been achieved. Papers on verifying code transformation during compilation and on interprocedural speculative optimization techniques have been published by TU Berlin.*
    b. Improving the **support of compilers towards efficient and correct code generation** will be the goal for the second year. Additional transformations will be considered in the verification effort.

6. **Transversal cluster**
   a. The activity maintains a link to the transversal cluster. *This goal has been achieved. The activity leader participated in the meeting at Rome on Nov. 12th and 13th.*
   b. The activity will maintain a **link to the transversal** cluster in the coming years. Cooperation will be on a partner by partner basis.

3.3 Main Funding

Main sources of funding are:

- **RWTH Aachen**
  The ISS institute at RWTH Aachen University also receives funds from
  - Deutsche Forschungsgemeinsschaft (DFG), e.g. via the new Excellence Cluster UMIC (Ultra High Speed Mobile Information and Communication), a large scale next-generation mobile internet research program.
  - EU FP6 projects like SHAPES, HiPEAC, and NEWCOM.
  - Industrial partners like Siemens, Nokia, Infineon, CoWare, and Tokyo Electron.
- **TU Dortmund**
  The group works on several projects, including
  - the MORE project aiming at middleware design for group communication http://www.ist-more.org
  - the PREDATOR project targeting predictable designs http://www.predator-project.eu
  - the MNEMEE project, which is performed at Dortmund's technology transfer center ICD. MNEMEE focuses on the exploitation of the memory hierarchy. http://www.mnemee.org
  - University funding
- **IMEC, Leuven**
  IMEC works on many projects, including
  - **MOSART IST-215244 Project:**
    Mapping Optimization for Scalable multi-core ARchiTecture. ArtistDesign partners involved: Interuniversitair Micro-Elektronica Centrum (IMEC) vzw., Democritus Uni. Thrace (DUTH) and Kungliga Tekniska Hagskolan (KTH) http://www.mosart-project.org/
  - **MNEMEE IST-216224 Project:**
    Memory maNagEMEnt technology for adaptive and efficient design of Embedded systems. ArtistDesign partners involved: Interuniversitair Micro-Elektronica Centrum (IMEC) vzw., Democritus Uni. Thrace (DUTH) and Technische Universiteit Eindhoven (TU/e) http://www.mnemee.org/
  - **IMEC Apollo research program:**
    Disruptive technologies needed to realize nomadic embedded systems for 2012 and beyond. These are technology aware architectures, multiprocessor system-on-chip technology, and reliable design methodologies for sub-45nm unreliable components. For the Apollo research, IMEC cooperates with industrial partners, such as integrated device manufacturers, fabless and fablite IC solution

providers, and system integrators.
http://www2.imec.be/imec_sites/objects/80acd42f851591023f893a7d96fd96bf/annualreport.pdf (page 36)

- **University of Passau**
  The group works on several projects, including
  - the CompSpread project supported by the Deutsche Forschungsgemeinschaft (DFG)
  - the FeatureFoundation project supported by the Deutsche Forschungsgemeinschaft (DFG). This project related to aspect-oriented programming is expected to find applications for embedded system programming.
  - the CoreGRID network of excellence
  - University funds

4. Internal Reviewers for this Deliverable

Prof. Dr. Olaf Spinczyk, TU Dortmund, Informatik 12