1 Chair of Computer Science 2 (Programming Systems)

Address: Martensstr. 3, 91058 Erlangen
Phone: +49 9131 85 27621
Fax: +49 9131 85 28809
E-Mail: info@i2.informatik.uni-erlangen.de

Head:
Prof. Dr. Michael Philippsen
Honorarprofessoren:
Hon.-Prof. Dr.-Ing. Bernd Hindel
Hon.-Prof. Dr.-Ing. Detlef Kips
Professor Emeritus:
Prof. em. Dr. Hans Jürgen Schneider
Secretary:
Agnes Brütting
Waltraud Rück
Scientific Coworkers:
Dipl.-Inf. Alexander Dreweke, B. Sc. mult.
Dipl.-Ing. (FH) Thorsten Edelhäußer (since May 1, 2008)
Dipl.-Inf. Ralf Ellner (since October 1, 2008)
Dipl.-Inf. Philipp Janda
Dipl.-Inf. Michael Klemm (until August 31, 2008)
Dr.-Ing. Norbert Oster (since October 1, 2008)
Dipl.-Inf. Dominic Schell (until March 31, 2008)
Ronald Veldema, Ph.D.
Dipl.-Inf. Tobias Werth
PD Dr.-Ing. habil. Peter Wilke
Dipl.-Inf. Marc Wörlein
Guests/Scholarship holders:
Dipl.-Inf. (FH) Josef Adersberger
Dipl.-Inf. Johannes Drexler
PD Dr.-Ing. Gabriella Kókai
Dipl.-Inf. Johannes Ostler
Dipl.-Inf. Stephan Otto
Dr. Karsten Schmidt
Dipl.-Inf. Szilvia Zvada
Externes Lehrpersonal:
Dr.-Ing. Klaudia Dussa-Zieger
Dr. Georg Heidenreich

Technical/Administrative Staff:
Dipl.-Ing. (FH) Helmut Allendorf
Manfred Uebler

The Chair of Computer Science 2 was founded in 1972 and is now headed by Prof. Michael Philippsen (as the successor of Prof. H.-J. Schneider) since April 2002. The staff consists of several scientists working on **programming systems** and three additional professors, one professor for **didactics of computer science** (see separate description) and two honorary professors leading the **practical software engineering group**.

1.1 Focus of research

The focal points of the **programming systems research group** are programming systems for both parallel and distributed systems as well as for embedded and mobile systems. Software (and its creation) for these platforms should not be more complex than regular desktop software for single processor systems. It should also be as portable, maintainable, and robust. Our long-term aim is to deliver the full performance of the hardware (both computing and communication performance) to the application, in spite of a high level of abstraction that is needed for the engineering of demanding applications.

In the area of **embedded systems** we have mainly worked on reducing the amount of memory required for the code by means of clever compiler optimizations. In the area of **parallel systems** we have continued to improve their programmability in 2008. We have worked on (homogeneous) clusters as well as grids that are composed of heterogeneous nodes. In 2008 we have contributed to programming systems for multicores, especially parallelization and optimization for multicore architectures.

The honorary professors Dr. Bernd Hindel and Dr. Detlef Kips are both general managers of medium size software companies and have a long term experience in industrial software projects. They lead the **Practical Software Engineering research group** that investigates scientific methods to develop complex software systems based on well-defined processes. The research focus is on systematic descriptions, modelling and evaluation of software development processes and their tool based application to real world problems. Adequate meta-models, notations and evaluation methods are analyzed and their practical applicability is assessed.

1.2 Research projects

1.2.1 Reparallelization and Migration of OpenMP Applications

Project manager:
Prof. Dr. Michael Philippsen
Project participants:
Dipl.-Inf. Michael Klemm
Ronald Veldema, Ph.D.
Duration: 1.11.2003–31.12.2008
Contact:
Prof. Dr. Michael Philippsen
Phone: +49 9131 85-27625
Fax: +49 9131 85-28809
E-Mail: philippsen@informatik.uni-erlangen.de

Grid computing opens up a new computing infrastructure and users gain access to a vast landscape of computing resources that are spread over the world and that are instantly accessible through the Internet. In the future, computational Grids are expected to deliver high-end computing power at the user's finger tips.

Today, typical users of a computational Grid only exploit a single cluster and are faced with the job schedulers that assign computing resources to applications. Job schedulers expect a resource estimation to exclusively allocate parts of the computing system for the application. Users typically over-estimate the resource limits of their applications to avoid premature termination of the application by the job scheduler. Over-estimating resource limits has negative effects on both the clusters' schedules as well as on the waiting time until an application eventually executes.

This research project climaxed in a completed PhD thesis that presents a solution alleviating the need of estimating the resource limits for OpenMP applications. A reparallelization of OpenMP applications is automatically computed, when new computing resources become available or are withdrawn. The application can be migrated between clusters of the Grid if an allocated resource is about to be exceeded.

Programmers do not need to change the application to enable reparallelization and migration. Instead, our ompiler transparently prepares the applications for reparallelization and migration. It adds code to the application to enable reparallelization and augments the application with a platform-independent, coordinated heckpointing algorithm for migration. A prototype implementation of a migration framework automatically discovers free resources and migrates the application to these resources. Measurements show that the overhead of reparallelization and migration is roughly 4 %, which we consider a negligible cost compared to the gain of flexibility.

1.2.2 Graph-based Procedural Abstraction

Project manager:
Prof. Dr. Michael Philippsen
Project participants:
Dipl.-Inf. Marc Wörlein
Dipl.-Inf. Alexander Dreweke, B. Sc. mult.
Dipl.-Inf. Tobias Werth
Start: 1.4.2006
Contact:
Prof. Dr. Michael Philippsen
Phone: +49 9131 85-27625
Fax: +49 9131 85-28809
E-Mail: philippsen@informatik.uni-erlangen.de

Embedded systems are even nowadays programmed in a machine-oriented fashion. The high level of abstraction and comfort one is used to in the development of desktop applications (object-orientation, garbage collection, exception handling, parallelism, aspect orientation, etc.) seems to be far away for embedded systems. Thus portability, robustness, and maintenance of the software are substantially impaired. This is also an important economical problem because Europe is not yet dominated by the USA in this field. So the long term goal must be to gradually increase the level of abstraction for programming embedded systems, i.e. optimization techniques are needed that achieve small and energy efficient code in spite of the increased abstraction level.

Apart from the obvious question how the well-known standard compiler optimization techniques affect code size, new specific questions arise for embedded systems. While the RAM consumption of an application plays hardly a role on desktop systems, it is crucial for embedded systems. Object-oriented code and especially libraries offer a substantial and mostly unused potential to reduce code size by means of procedural abstraction. Not only code size, but energy efficiency is a target for code optimization on embedded systems. Possibly in cooperation with the operating system, compiler optimizations are crucial. A challenging problem is how to deal with the non-uniform memory hierarchy, that not only consists of registers, caches, and main memory. But in addition, there is another layer, e.g. Flash memory. Another question for embedded systems is wether they can be programmed with the illusion of a uniform memory hierarchy. Is it possible to extract information about data locality through statical analysis? What kind of optimizations are practicable with this information? Can statical analyzes and runtime mechanisms benefit from each other in this specific area? How can prefetch and post-store commands be generated in the code that help improve latency and energy

consumption?

A common method to reduce the size of program code is procedural abstraction (PA): repeated code portions in a program are identified and moved into a single new procedure. Instead of the extracted code portions, calls to the newly created procedure are inserted. This removes redundancy from the program and thus reduces its size. Earlier PA algorithms regarded a program as a sequence of instructions and searched for subsequences that appear frequently. However, if the instruction sequence is altered within a subsequence, they are no longer recognized by sequence matching algorithms. Hence, the result is suboptimal. To solve this problem we transform the instructions of a basic block into a data flow graph (DFG) and use a graph mining tool to search for common fragments in the DFGs of ARM assembly codes that are often used in embedded systems. In cooperation with the project ParSeMiS that is concerned with general optimizations of graph miners, specific properties of the PA domain are exploited. Our research has focused mainly on the analysis optimization of the correct reconstruction of data-flow graphs. The more accurate the data-flow analysis is, the more size reduction can be archived, compared to the traditional sequential approaches. In addition, various extraction mechanisms have been refined. These mechanisms are used to extract code fragments that have been identified to be frequent by ParSeMiS. The designed extraction mechanisms are as size-efficient as possible and standardize semantic equivalent fragments in a way that they can be extracted together. To speed up the search for frequent code fragments, we have started research on various heuristics in 2008.

1.2.3 Compiler-supported parallelization for multi-core architectures

Project manager:
Prof. Dr. Michael Philippsen
Project participants:
Dipl.-Inf. Tobias Werth
Dipl.-Inf. Dominic Schell
Start: 1.3.2007
Contact:
Dipl.-Inf. Tobias Werth
Phone: +49 9131 85-28865
Fax: +49 9131 85-28809
E-Mail: werth@informatik.uni-erlangen.de

Several issues significantly retard the development of quicker and more efficient computer architectures. Traditional technologies can no longer contribute to more hardware speed. Basic problems are the divergent ratio of the latencies of memory

access and CPU speeds as well as the heat and waste of energy caused by increasing clock rates.

Homogeneous and heterogeneous multi-core architectures were presented as a possible answer and offer enormous performance to the programmer. The decreasing clock rates help avoid most of the above problems, while the multiplied hardware can still deliver high performance since more arithmetic operations can be executed per time unit with less energy. Potentially, performance can increase even further by specialization of some hardware components. For example, often the latency problem is attacked with a multi-tiered memory hierarchy and lots of caches.

But there is no free lunch. It seems to be quite difficult to make multi-core architectures deliver their theoretically available performance to applications. Only with a lot of expertise in both the application domain and the specifics of the multi-core platform at hand and only with enough time to invest into tuning endeavors, one can make multi-core programs run fast.

From the point of view of a programming systems research group, there are - among others - the following open questions: What kind of support can a modern compiler offer to the programmer that develops applications for multi-core architectures? How much context knowledge is necessary in order to make reasonable decisions for parallelization? Which part of the available performance can be uleashed by the programmer with a reasonable amount of effort without detailed knowledge about the features and quirks of the underlying architecture? Which tools are necessary for debugging and for finding bottlenecks in applications that run on multi-core architectures? How can they be designed?

It is the intention of this research project to answer these questions for a restricted application domain. We have selected the Lattice-Boltzmann-Method (LBM) that is mostly used in computational fluid dynamics as our problem domain. Caused by its lattice structure and its manageable number of data dependencies between the single lattice points, it is comparatively straightforward how to parallelize it. Hence, our compiler research can focus on the above questions.

In 2008, our research had multiple topics:

- To gauge the related work of our project, we compared several programming models for the CellBE architecture, investigated their advantages and disadvantages, and compared their runtimes.
- Furthermore, we worked on a design for the programming model Cilk that was originally designed for homogeneous multi-core architectures with shared memory. In contrast, our design focused on the CellBE architecture that is heterogeneous and does not share local memories. Our design extends Cilk with few

keywords in order to use the parallel performance of the co-processors (SPUs). Therefore, the code is translated by a source-to-source-compiler and annotated regions are executed in separate threads, if possible and necessary.

- The SPUs can access only their small local store (256kB) directly. This space has to be used for code, stack, and data. The idea of our program loader is as follows: not the whole program code is executed at the same time in order to execute the program properly. The idea of this approach results of an ongoing dissertation (completion in 2009) that investigates analogous research topics for ARM processor with scratchpad memories. In this framework, the loader fragments the code dynamically at runtime (from basic blocks to functions) and loads these fragments into a code cache. If memory is low within the code cache, a special purpose garbage collector automatically removes currently unused code fragments. Benchmarks showed that the loader can save up to 70 % of the code space compared to native execution with a relative low runtime overhead of 5 % on average.

1.2.4 Tapir

Project manager:
Prof. Dr. Michael Philippsen
Project participants:
Ronald Veldema, Ph.D.
Dipl.-Inf. Michael Klemm
Duration: 1.1.2006–31.12.2010
Contact:
Ronald Veldema, Ph.D.
Phone: +49 9131 85-27622
Fax: +49 9131 85-28809
E-Mail: veldema@informatik.uni-erlangen.de

Tapir is a new programming language for easier systems programming.

Systems programming includes networking protocols, operating systems, middlewares, DSM systems, etc. Such systems are critical for the functioning of a system as they supply services that are required by user applications. For example, an operating system supplies an operating environment and abstracts from concrete hardware in doing so. A DSM system simulates a single shared memory machine by abstracting from the single machines inside a cluster so that a (user-level application on a) cluster can be programmed without having to program explicit message passing.

Compared to application programming, systems programming has a different set of requirements. Also, the programming 'style' is very different from the styles used in programming user-level applications. Finally, the performance requirements are usually very strict in systems programs as the complete system's performance greatly relies on the underlying layers of systems programs. Also, bugs in systems code have great repercussions on a complete system's reliability. Combined, we can directly conclude the following when using high-level languages for systems programming:

- High-level languages, such as C++, C#, and Java 'hide' implementation details from the programmer. A programmer for example no longer needs to know how exactly a method call is implemented. This knowledge is, however, required when doing systems programming.
- High-level languages supply functionality that is neither required nor wanted. For example, when programming an operating system, automatic language driven exception handling or garbage collection are not wanted.
- Systems programs require no high abstraction levels like common high-level programming languages supply. Likewise, the libraries that a given language offers can simply not be supplied in a systems context. Usually this is due to the system itself supplying the functionality that the library is to provide.

The basics of the Tapir language have been created. While Tapir has some similarities to Java, C#, and C++, all unneeded and unwanted functionality of the above have been removed. For example, Tapir has no automatic memory management, no exception handling, and no type-casts. Class and object concepts have been kept, but inheritance has been removed. The resulting Tapir programs can be verified by means of model-checking, even while the programming is still being developed. A prototype of the Tapir compiler and the Tapir model-checker is operational. While the language is still being developed, a prototype DSM protocol has already been implemented in the Tapir language. We have evaluated RDMA-based DSM protocols so that they can be added to the Tapir language. Tapir's semantic checks are implemented by means of model-checking. Model-checking, however, is a very memory intensive analysis. This made us write our own Java Virtual machine, called LVM, which is especially suited for managing large numbers of objects. LVM outperforms standard Java VMs as soon as swapping becomes necessary.

In 2008, LVM was optimized both for sequential execution and for distributed execution on a cluster of workstations. This allows for faster verification of Tapir programs on clusters and for faster running of scientific Java programs.

1.2.5 JavaParty

Project manager:
Prof. Dr. Michael Philippsen
Project participants:
Dipl.-Inf. Marc Wörlein
Start: 1.4.2007
Contact:
Prof. Dr. Michael Philippsen
Phone: +49 9131 85-27625
Fax: +49 9131 85-28809
E-Mail: philippsen@informatik.uni-erlangen.de

[JavaParty]http://svn.ipd.uni-karlsruhe.de/trac/javaparty/wiki/JavaParty allows easy port of multi-threaded Java programs to distributed environments such as clusters. Regular Java already supports parallel applications with threads and synchronization mechanisms. While multi-threaded Java programs are limited to a single address space, JavaParty extends the capabilities of Java to distributed computing environments.

The normal way of porting a parallel application to a distributed environment is the use of a communication library. Java's Remote method invocation (RMI) renders the implementation of communication protocols unnecessary, but still leads to increased program complexity. The reasons for increased complexity are the limited RMI capabilities and additional functionality that must be implemented for creation and access of remote objects.

The JavaParty approach is different. JavaParty classes can be declared as remote. While regular Java classes are limited to one Java virtual machine, remote classes and their instances are visible and accessible anywhere in the distributed JavaParty environment. As far as remote classes are concerned, the JavaParty environment can be viewed as a Java virtual machine that is distributed over several computers. The access and the creation of remote classes is syntactically indistinguishable from regular Java classes.

In 2008, a complete new version of the JavaParty compiler was implemented. This version supports the current Java Standard 1.5/1.6. The implementation is based on the open and freely available Eclipse compiler framework. Thus, future developments of the Java language and corresponding extensions for the Eclipse compiler will automatically become available for JavaParty as well.

1.2.6 ParSeMiS - the Parallel and Sequential Mining Suite

Project manager:
Prof. Dr. Michael Philippsen

Project participants:
Dipl.-Inf. Marc Wörlein
Dipl.-Inf. Alexander Dreweke, B. Sc. mult.
Dipl.-Inf. Tobias Werth
Start: 1.5.2006
Contact:
Prof. Dr. Michael Philippsen
Phone: +49 9131 85-27625
Fax: +49 9131 85-28809
E-Mail: philippsen@informatik.uni-erlangen.de

The **ParSeMiS** project (**Par**allel and **Se**quential Graph **Mi**ning **S**uite) searches for frequent, interesting substructures in graph databases. This task is getting increasingly popular, because science and commerce need to detect, store and process complex relations in huge graph structures.

For huge data that cannot be worked on manually, algorithms are needed that detect interesting correlations are necessary. Since in general the problem is NP-hard and requires huge amounts of computation time and memory, parallel or specialised algorithms and heuristics are required that can perform the search within time boundaries and memory limits.

Our target is to provide an efficient and flexible tool for searching in arbitrary graph data, to improve the adaption to new application areas, and to simplify and unify the design of new mining algorithms.

In 2008, the following goals have been achieved:

- Documentation and publication of the source code to enlarge the user base of the project,
- Implementation of a specialized graph layout for DAGs,
- Restructuring of the graphical user interface, and
- Added support for clusters of multi-core nodes.

1.2.7 Model Driven Component Composition

Project manager:
Prof. Dr. Michael Philippsen
Project participants:
Dipl.-Inf. Philipp Janda

Duration: 15.6.2007–14.6.2010
Sponsored by:
AUDI AG

This Ph.D. project started 2007 as part of the INI.FAU collaboration between AUDI AG and the University of Erlangen-Nuremberg. It will examine model-driven ways to integrate vehicle functions on electronic control units (ECUs). Moreover, the project will develop supporting methods and tools for this task. The insights gained in the course of this project will be practically validated by integrating a damper control system into an AUTOSAR ECU.

In the automotive industry it is common practice to develop in-car-software on a high level of abstraction and in a model-based way. To eliminate uncertainties concerning resource consumption and runtime it is necessary to test the developed software on the target hardware as early as possible. But due to cost and safety requirements the integration of the software into an ECU is very time-consuming and demands special expertise going beyond that of the function developer. AUTOSAR (AUTomotive Open Systen ARchitecture) is on the way to become a standard for the basic software on ECUs. But due to the novelty of this standard there are neither processes nor tools to support the integration of the developed in-car-software into an ECU.

In 2008, we have examined the modeling expressiveness of AUTOSAR with respect to both its applicability and possible conflicts with existing standards and technologies that are currently in use at Audi. Furthermore, the automatic generation of an AUTOSAR software architecture from a single damper control component has been implemented. Future work will focus on the automatic configuration of the bus communication and the task scheduling among the application processes.

1.2.8 Integrated Tool Chain for Meta-model-based Process Modelling and Execution

Project manager:
Hon.-Prof. Dr.-Ing. Detlef Kips
Project participants:
Dipl.-Inf. Ralf Ellner
Prof. Dr. Michael Philippsen
Dr.-Ing. Martin Jung
Dipl.-Inf. Johannes Drexler
Samir Al-Hilank
Duration: 1.10.2008–30.9.2011
Sponsored by:

BMWi
Participating institutions:
develop group, Erlangen

As demands on the development of complex software systems are continuously increasing, compliance with well-defined software development processes becomes even more important. Especially large and globally distributed software development projects tend to require long-running and dynamically changeable processes spanning multiple organizations. In order to describe and support such processes, there is a strong need for suitable process modeling languages and for powerful support by tools.

Within the reporting period, a cooperation project funded by BMWi has been carried out together with develop group as an industrial partner, in order to evaluate the capabilities of several process modeling languages, especially of the Object Management Group's (OMG) Software and Systems Process Engineering Metamodel (SPEM). Furthermore, in this cooperation project the functional properties of various existing commercial software development process management tools have been captured and analyzed. Some of the evaluation results were presented at the Software Engineering Conference 2008 in Munich.

The results of that cooperation project show that todays tools markets lack integrated tool chains which actually support the fine-grained and precise modeling of software development processes as well as their computer-aided execution, controlling and monitoring. Another cooperation project, which intends to bridge this gap, has been started recently.

The objective of this cooperation project is to prototype an integrated tool chain by using a rigorous, metamodel based approach which supports modeling, enactment and execution of industrial software development processes. Bearing the applicability of such a tool in mind, this approach is mainly intended to provide a wide adaptability of process models to different industrial development scenarios, to define a user-friendly concept of process description and to establish an extensive computer-aided process execution support, contributing to the efficiency of development activities. These benefits will be achieved by a high grade of formalism, by an integrated, generic concept of process modeling and process enactment and by using commonly accepted industrial standards (UML, SPEM).

This ongoing cooperation project will also be carried out together with develop group as an industrial partner and will be funded by BMWi, too. It started in October of 2008 and has been scheduled for three researchers over a period of three years.

1.2.9 Wireless Localization of Antenna Positions

Project manager:
PD Dr.-Ing. Gabriella Kókai
Project participants:
Dipl.-Ing. (FH) Thorsten Edelhäußer
Duration: 1.5.2008–30.4.2011
Sponsored by:
Fraunhofer Institut für Integrierte Schaltungen

In 2008, we have developed a software to estimate the position of the receiving antenna of a localization system. We used robots to determine the position and direction of the antenna. The robot determines the characteristic features of the localization system and stores the data in a database. Our developed algorithm uses this data to estimate both the position and the direction of the antenna with the help of heuristic optimization algorithms. Experiments in a real scenario confirm the usability of our method in real applications.

1.2.10 Evolutionary agents

Project manager:
Dipl.-Inf. Stephan Otto
Start: 1.4.2008
Contact:
Dipl.-Inf. Stephan Otto
Phone: +49 9131 85-27830
Fax: +49 9131 85-28809
E-Mail: Stephan.Otto@informatik.uni-erlangen.de

There is a trend towards complex and distributed systems, complicating the design process of self-adaptive and complex software systems. Examples are the internet, grid and peer-to-peer based systems and subsequent developments like multi agent systems. The growth of the World Wide Web and the rapid rise of eCommerce have led to significant efforts to develop distributed and complex software and technologies to support and enable the engineering of systems involving distributed computation. These effects originated from inherent distributed problems e.g. global supply chains or collaborative internet based applications. As a consequence this development has led to more decentrality which has a number of practical advantages over centralized approaches:

- Central elements are limited by how much they can communicate, store and process,
- Centralized solving may be infeasible due to privacy and data integration problems,
- Dynamic environment: by the time we manage to centralize the problem, it has already changed.

In this context of dynamic, inaccessible and distributed structures the adaptation and optimization of systems and processes still poses problems. Current adaptation or optimization techniques rely on central processing. There exist a number of approaches to tackle problems bottom-up but they require cooperative and predefined behaviour of involved actors and such methods work mostly on a limited set of problems. Therefore in current adaptation and optimization methods exist a number of problem areas which need more investigation: distribution, heterogeneity and dynamic environments.

This project focuses on a tolerant approach towards the previously mentioned aspects of investigation. Whilst the top-down approach tackles problems by programming intelligence from above, the bottom-up approach tries to build structures as a result of specifying a set of simple rules, a set of simple elements, called agents, which adhere to those rules. A main characteristic of complex systems is their bottom-up approach on adaptation and self-organization without following central directions. This project aims to develop an approach that provides adapting and self-optimizing decentralized applications using a multi-agent system (MAS) based on Evolutionary Computation (EC). Our approach uses fully decentralized operators for reproduction like mutation, recombination and selection, regulated by market mechanisms. This tackles the existing bottleneck of selection and fitness comparing in current distributed evolutionary approaches. The novelty of this approach lies in the decentralized bottom-up adaptation and optimization method for decentralized systems and is applied to various scenarios. Our proposed method is based on a formal model explaining how adaptation occurs in the number and strategies of agents and thus of the emergent adaptation and optimization itself.

The key contribution of the present project is summarized as follows:

- Development of a new distributed evolutionary algorithm. Central selection and fitness comparison are omitted,
- Development of endogenous fitness and their effect on the quality of solutions,
- Development and comparison of local selection methods,

- To put the results on a solid basis, investigations on the takeover time on weakly connected graphs have been conducted to evince appropriate information flow in weakly connected, distributed and dynamic structures,
- A new measure of decentrality has been developed to help classify distributed approaches,
- Several scenarios were successfully tested using this approach.

1.2.11 Optimization of FIR filter structures

Project manager:
PD Dr.-Ing. Gabriella Kókai
Project participants:
Dipl.-Inf. Szilvia Zvada
Dipl.-Ing. Hans Holm Frühauf
Duration: 1.1.2006–30.9.2009
Contact:
PD Dr.-Ing. Gabriella Kókai
Phone: +49 9131 85-28996
Fax: +49 9131 85-28809
E-Mail: kokai@informatik.uni-erlangen.de

Due to the boom of electronic systems in everyday life, VLSI (very large scale integration) circuits quickly became the hot-spot of intensive research. The main issue in this area is to design chips that are small, fast, and consume little power. The advancement of modern chip fabrication technology and the increasing packing density have made it possible that today's VLSI chips contain a few million transistors. From a chip designer's point of view this unfolds a vast number of possible filter structures while looking for an optimal or near-optimal chip. Thus, the automation of the design process becomes increasingly important.

In case of digital filters much attention has been paid to the design of the FIR (finite impulse response) filters. These filters are widely used to transform digital data sequences according to some specific linear function, e.g. the linearization of power amplifiers or the calibration of audio or video receivers. If, however, a non-linear transformation of the data sequences is required, a great part of the design process must be done manually.

The evolFIR system developed in this project fills this gap by providing a sophisticated tool that can optimize the logic design of polynomial FIR filter structures. At this stage of the circuit design, functional block elements (adders and multipliers) and logical primitives (shifts and delays) are composed in order to fulfill the required functionality

of the target filter. This functionality is specified by the filter's transfer function. During the evolutionary process, we firstly have to ensure that the individuals always describe the desired transfer function. Secondly, we have to assure that the composed topology fulfills certain hardware-related requirements known from the subsequent levels of chip design, such as the limited number of input channels or the restricted word-width of the dedicated block element.

The main objective of evolFIR is to obtain a redundancy-free filter design with as few block elements as possible. We achieve this by employing AGGP (attribute grammar based genetic programming) as follows:

- The individuals of the evolutionary process are special derivation trees that represent possible topologies of the available functional elements. Since we allow the usage of multipliers in the compositions, polynomial transfer functions can also be dealt with.
- By means of attributes and the smart random tree generator of AGGP, exactly those derivation trees are surveyed during the optimization that represent filter descriptions of the given transfer function.
- Redundant components of the filter descriptions are eliminated by using a unique representation form of the derivation trees: the abstract-linked derivation trees. Hence, only the redundancy-free filters are included in the search space.
- In addition, the random tree generator is enhanced to take the hardware-related constraints into consideration: it only creates such derivation trees that fulfill these constraints.

Altogether, the evolFIR method operates on an inhomogeneous search space due to these various restrictions. The inhomogeneity of the search space and our particular representation form imply that the fitness function is non-continuous. Therefore, the evolutionary engine requires careful parameterization in order to avoid premature convergence of the evolutionary process. In 2008, we have studied the interaction of the evolutionary parameters and the hardware-related constraints and parameters, as well as their particular impact on the filter optimization problem.

1.2.12 Time-Tabling Algorithms

Project manager:
PD Dr.-Ing. habil. Peter Wilke
Project participants:
Dipl.-Inf. Johannes Ostler

Duration: 1.1.2004–31.12.2010
Contact:
PD Dr.-Ing. habil. Peter Wilke
Phone: +49 9131 85-27624
Fax: +49 9131 85-28809
E-Mail: wilke@informatik.uni-erlangen.de

Time tables have to be generated in quite different areas, e.g. human ressources, school time tabling etc. Because scheduling is a time consuming task, especially if the context is complex, most time tables are generated by computer programs. We have developed a sophisticated software which enables us to generate optimised time tables using different optimisation algorithms in recent years. The current software is a complete new implementation as the former version of the software showed some design flaws which persuaded us to re-design the architecture.

Erlangen Advanced Time Tabling Software EATTS is the innovative development and production environment to generate optimized time tables.

Resources

Time tabling problems are quite common and come in different versions, among them rosters, schedules and school time tables. They have in common that given resources have to be used as efficient as possible and that this requires planning with respect to the given constraints to obtain a decent plan. When looking at a school time table the events are lessons, to which the resources like teachers, classes ans rooms have to be assigned. All resources are typed. Each type has as many attributes created by the user as required.

Planning a time table usually begins with compiling the resources, either by reading in a file or typing in them manually. The screen shot shows the input dialogue to enter the attribute values for a class. All resources of the type ”Class” have to be assigned values for the user defined attributes ”name”, ”size”, ”grade” and ”room”.

Results

Time tables are the output of the planning algorithms and can be stored in different file formats and views. The screen shot shows the view of a student, e.g. he sees his personal time table consisting of the lessons he has to attend to. Other views can be created instantly, for a teacher, a headmaster or a caretaker. All users access the EATTS through a browser providing an interface according to the privileges of the user.

A common view indicates which constraints are currently not satisfied naming the events and resources. Based on this information the administrator can decide if he would like to edit the resources, events or constraints or use a different algorithm. The screen shot show time slot clashes and therefore the algorithm should be given more time to find a solution a another algorithm should be given a shot.

Constraints

The specification of constraints is usually more complex than describing resources or events.

On one hand a precise specification is required and one the other hand the current setting should be presented clearly to find gaps and/or inconsistencies, which can't be avoided automatically.

Constraints come in different flavours, therefore a flexible way to specify them is necessary. EATTAS allows to refer to resources, their classes and all attributes. Depending on the type of the attributes, among them are integer, double and string, arithmetic and logical operators can be used to specify the constraint. In addition the parameters of the cost function are set, to compute the correct penalty point if the constraint is violated.

A unique property of EATTS is the option to specify a constraint not only as "has to be fulfilled (hard)" or "should be fulfilled (soft)" but also as "can be violated in exceptional case (soft hard)". This allows to violate constraints when it is acceptable, e.g. a room isn't available due to a broken water pipe, a teacher isn't available due to a traffic jam. In these cases a time table should be created which is similar to the one currently in effect but violates some constraints to minimize changes.

Running Experiments

The algorithms can be executed on a dedicated server or can be distributed over a TCP/IP network on additional computers. The screen shot shows the dialogue where the user can select the experiments and start their execution. The browser contacts the server regularly and updates the status information, including the costs of the best plan found so far and an estimated remaining computation time. At the end of the computation results are stored and the data required for the views are generated. Now the user decided whether the results are satisfactory or additional computations are required.

Results

Time tables are the result of the application of algorithms to the specifies resources, events and constraints. Time tables can be stored and displayed in various formats, enabling the display of different views of the time table.

Users typically access the EATTS via a browser and views are created depending on the users privileges.

Summary

The software is implemented in Java and available for numerous platforms, among them Windows and Linux operating systems.

To run EATTS the following free-ware software products are required:

- Java Runtime Environment (JRE), v5.0 are above

- computers connected via TCP/IP, if additional computing power is required (optional)

2008

In 2008 the structure of the algorithms was optimized to enhance their parallel computation. We will look into the use of multi core processor in the near future.

As attempts to install the software by users have shown that this might be too complex a downsized version which does not need a database but uses XML documents to store and exchange data was implemented. In addition a server was set up running on a computer at the University of Erlangen allowing the users to run their experiments on this mashine.

The user interface was reimplemented and is now a web based application.

At CeBIT 2009 the new version of the software will be published. And it was named EATTS Erlangen Advanced Time tabling System.

1.2.13 Graphs and Graph Transformations

Project manager:
Prof. em. Dr. Hans Jürgen Schneider
Duration: 1.10.2004–30.9.2010
Contact:
Prof. em. Dr. Hans Jürgen Schneider
Phone: +49 9131 85-27620
Fax: +49 9131 85-28809
E-Mail: schneider@informatik.uni-erlangen.de

Graphs are often used as an intuitive aid for the clarification of complex matters. Outside the field of computer science this is for example true in biology or chemistry, where molecules are modeled in a graphical way. In computer science data or control flow charts are often used as well as entity relationship charts or Petri-nets to visualize software or hardware architectures. Graph grammars and graph transformations combine ideas from the fields of graph theory, algebra, logic, and category theory, to formally describe changes in graphs.

The underlying theory is an attractive tool for the description of extremely different structures in a uniform way, e.g., the different models for asynchronous processes: Petri-Nets are based on standard labeled graphs, state charts use hierarchical graphs, parallel logic programming can be interpreted in a graph-theoretical way using so-called jungles, and the actor systems can be visualized as graphs, whose labeling alphabet is a set of term graphs.

In 2008, we have concentrated our attention on a theoretical concept as well as on an implementation aspect:

- A data base transaction is described by a sequence of atomic transformations, but from outside, it should be considered an indivisible step. From the graph transformation point of view, the question is: If we have a derivation sequence, can we construct a production simulating the effect of the whole derivation sequence by one derivation step? The problem has been already studied in our seminal paper (1973), but at that time, we assumed all the morphisms to be injective graph morphisms. In 1976, H. Ehrig and H.-J. Kreowski extended the solution to labeled graphs, and in 1977, H. Ehrig admitted noninjective embeddings, but his solution is based on set-theoretic arguments and doesnot use the categorical constructions. In 2006, H. Ehrig et al. have presented a purely categorical treatment, but this presentation considers only the special case of adhesive categories and is also restricted to monomorphisms. Now, we could show that these restrictions are too strong. Our solution is purely categorical, too, but does not impose any restrictions to the morphisms. Furthermore, we need only that the category under consideration has pushouts and pullbacks, i.e., we can apply the result to any category of interest, e.g., to the category of structurally labeled graphs that is not adhesive. The proof can be found in the draft of our textbook: http://www2.informatik.uni-erlangen.de/Personen/schneide/gtbook/chapter5.pdf

- The categorical approach to graph transformations is highly generic: All the proofs and constructions are valid for various types of graphs. Only the basic operations must be described for each application in detail, the categorical properties on top of these are then defined automatically. Since modern programming languages support generic concepts, it is a promising idea to implement the categorical approach to graph transformation in languages like Java or Haskell. Java uses classes of objects, but does not really support multiple inheritance, since method definitions are not allowed in interfaces. Therefore, we need factory classes to implement generic categorical constructions, and we have to import them into each implemented category. On the other hand, Haskell supports multiple inheritance, but considers classes of types and requires concrete types to be explicitly made instances of all the classes they belong to. From our pilot implementations, interesting questions arise concerning the differences in implementing genericity. The Haskell version is already available: http://www2.informatik.uni-erlangen.de/Personen/schneide/gtbook/appendix-a.pdf A summary of the Java version will be available in the near future. The main parts, however, can be found in the material accompanying the lecture on graph transformations systems: http://www2.informatik.uni-erlangen.de/Lehre/WS200809/GraTra/index.xml

1.2.14 International Collegiate Programming Contest at the FAU

Project manager:
Prof. Dr. Michael Philippsen
Project participants:
Dipl.-Inf. Tobias Werth
Dipl.-Inf. Marc Wörlein
Dipl.-Inf. Alexander Dreweke, B. Sc. mult.
Start: 1.11.2002
Contact:
Dipl.-Inf. Tobias Werth
Phone: +49 9131 85-28865
Fax: +49 9131 85-28809
E-Mail: werth@informatik.uni-erlangen.de

The Association for Computing Machinery (ACM) has been hosting the International Collegiate Programming Contest (ICPC) for many years. Teams of three students try to solve nine to ten programming problems within five hours. What makes this task even harder, is that there is only one computer available per team. The problems demand for solid knowledge of algorithms from all areas of computer science and mathematics, e.g. graphs, combinatorics, strings, algebra and geometry.

The ICPC consists of three rounds. First, each participating university hosts a local contest to find the three teams that are afterwards competing in one of the various regional contests. Erlangen lies in the catchment area of the Northwestern European Regional Contest ([NWERC]http://2009.nwerc.eu) where also teams from e.g. Great Britain, Benelux and Scandinavia. The winners of all regionals in the world (and some second place holders) advance to the world finals in spring of the following year.

In 2008 two local contests took place in Erlangen. During the winter semester a team contest was conducted with teams consisting of at most three students. The main goal of this contest was to interest new students in the contests. Also some teams from TU Munich and University of Konstanz competed with our teams online.

As in the previous years in the summer term the seminar ”Hello World - Programming for Advancers” served to prepare students for the contests. In the contest of the summer semester the representatives of the FAU Erlangen-Nuremberg for the SWERC 2008 in Lisbon were chosen. 25 students of computer science, computational engineering, mathematics as well as information and communication technology took part in the contest. Ten students were selected for the SWERC forming three teams and one reserve. At the SWERC in Nürnberg (hosted by the FAU!), the three Erlangen teams managed to finish on ranks 4, 10, und 26 of a total of 56 teams. Therefore, with one gold an one bronze medal, the trainings-camp was also in 2008 a great success.

It was the first time for the FAU to host an ICPC [regional contest]http://icpc.informatik.uni-erlangen.de/swerc2008/. Almost 60 teams from allover southwestern europe tried to become the SWERC-champion in Nürnberg. In 2009, the FAU will host again a regional contest but this time for northwestern Europe.

1.3 Publications

– Beyler, Jean Christophe ; Klemm, Michael ; Philippsen, Michael ; Clauss, Philippe: Automatic Prefetching with Binary Code Rewriting in Object-based DSMs (Best Paper) . In: Luque, Emilio ; Margalef, Tomàs ; Benítez, Domingo (Ed.) : EuroPar 2008 - Parallel Processing (Proceedings of the Euro-Par 2008 Conference Las Palmas de Gran Canaria, Spain 26.-29.08.2008). Vol. LNCS 5168. Heidelberg, Germany : Springer, 2008, pp 643-653. - ISBN 978-3-540-85450-0

– Brunner, Maximilian ; Jung, Martin ; Kips, Detlef ; Schmidt, Karsten: Fallstudie zur Modellierung von Software-Entwicklungsprozessen auf Basis von SPEM 2.0 . In: Herrmann, Korbinian ; Brügge, Bernd (Ed.) : Software Engineering 2008: Fachtagung des GI-Fachbereichs Softwaretechnik (Software Engineering 2008 München 18.-22.2.2008). Vol. 121. LNI : GI, 2008, pp 67-74. - ISBN 978-3-88579-215-4

– Dreweke, Alexander: Graph-Based Procedural Abstraction . Saarbrücken : VDM, 2008. - 60 pages. ISBN 978-3-8364-6568-7

– Dreweke, Alexander: Lattice Boltzmann Method for DSM Systems . Saarbrücken : VDM, 2008. - 56 pages. ISBN 978-3-8364-6567-0

– Drexler, Johannes: Untersuchung und Evaluation von Methoden, Notationen und Werkzeugen zur Modellierung von Software-Entwicklungsprozessen . Erlangen : Basys GmbH. 2008. - Final report (Bericht des Personalaustausch-Projektes KP0563901BN7A im Rahmen des Förderprogramms ProInno II)

– Klemm, Michael ; Veldema, Ronald ; Bezold, Matthias ; Philippsen, Michael: A Proposal for OpenMP for Java . In: Mueller, Matthias S. ; Chapman, Barbara M. ; de Supinski, Bronis R. ; Malony, Allen D. ; Voss, Michael (Ed.) : OpenMP Shared Memory Parallel Programming (International Workshops IWOMP 2005 and IWOMP 2006) (International Workshop on OpenMP Reims, France 12.-15.06.2006). Berlin, Germany : Springer, 2008, pp 409-421. - ISBN 3-540-68554-5

– Klemm, Michael ; Veldema, Ronald ; Philippsen, Michael: An Automatic Cost-based Framework for Seamless Application Migration in Grid Environments .

In: Gonzalez, Teofilo F. (Ed.) : Proceedings of the 20th IASTED International Conference on Parallel and Distributed Computing and Systems (20th IASTED International Conference on Parallel and Distributed Computing and Systems Orlando, FL, USA 16.-18.11.2008). Anaheim, CA, USA : ACTA Press, 2008, pp 219-224. - ISBN 978-0-88986-773-4

– Klemm, Michael ; Veldema, Ronald ; Philippsen, Michael: Cluster Research at the Programming Systems Group . In: High Performance Computing at RRZE (2008), pp 30-31

– Otto, Stephan ; Kókai, Gabriella: Decentralized Evolutionary Optimization Approach to the p-median Problem . In: Giacobini, Mario ; Brabazon, Anthony ; Cagnoni, Stefano ; Di Caro, Gianni ; Drechsler, Rolf ; Ekárt, Anikó ; Esparcia-Alcázar, Anna ; Farooq, Muddassar ; Fink, Andreas ; McCormack, Jon ; O'Neill, Michael ; Romero, Juan ; Rothlauf, Franz ; Squillero, Giovanni ; Uyar, A. Sima ; Yang, Shengxiang (Ed.) : Applications of Evolutionary Computing - EvoWorkshops 2008: EvoCOMNET, EvoFIN, EvoHOT, EvoIASP EvoMUSART, EvoNUM, EvoSTOC, and EvoTransLog (Joint Conferences on Evolutionary Computing (EuroGP EvoCOP EvoBio and EvoWorkshops) Naples, Italy 26.-28.03.2008). Berlin / Heidelberg : Springer Verlag, 2008, pp 659-668. (Lecture Notes in Computer Science Vol. 4974) - ISBN 978-3-540-78760-0

– Pinte, Florin ; Saglietti, Francesca ; Oster, Norbert: Automatic Generation of Optimized Integration Test Data by Genetic Algorithms . In: Maalej, Walid ; Brügge, Bernd (Ed.) : Software Engineering 2008 - Workshopband (Software Engineering 2008, Workshop "Testmethoden für Software - Von der Forschung in die Praxis" München 2008). Bonn : Gesellschaft für Informatik (GI) e. V., 2008, pp 415-422. (Lecture Notes in Informatics Vol. P - 122) - ISBN 978-3-88579-216-1

– Pinte, Florin ; Baier, Gerhard ; Saglietti, Francesca ; Oster, Norbert: Automatische Generierung optimaler modellbasierter Regressionstests . In: Hegering, Heinz-Gerd ; Lehmann, Axel ; Ohlbach, Hans Jürgen ; Scheideler, Christian (Ed.) : INFORMATIK 2008 - Beherrschbare Systeme dank Informatik (Band 1) (Workshop Modellbasiertes Testen München 09.09.2008). Vol. 1. Bonn : Gesellschaft für Informatik, 2008, pp 193-198. (Lecture Notes in Informatics Vol. P-133) - ISBN 978-3-88579-227-7

– Pinte, Florin ; Oster, Norbert ; Saglietti, Francesca: Techniques and Tools for the Automatic Generation of Optimal Test Data at Code, Model and Interface Level . In: ACM (Ed.) : ICSE Companion '08: Companion of the 30th International Conference on Software Engineering (ICSE 2008) (International Conference on Software Engineering (ICSE 2008) Leipzig). USA : ACM, 2008, pp 927-928. - ISBN 978-1-60558-079-1

– Saglietti, Francesca ; Oster, Norbert ; Söhnlein, Sven: Qualität und Zuverlässigkeit im Software Engineering . In: Zeitschrift für wirtschaftlichen Fabrikbetrieb (ZWF) 103 (2008), No. 6, pp 407-412

– Saglietti, Francesca ; Oster, Norbert ; Pinte, Florin: White and Grey-Box Verification and Validation Approaches for Safety- and Security-Critical Software Systems . In: Information Security Technical Report, Elsevier 13 (2008), No. 1, pp 10-16

– Veldema, Ronald ; Bradford, Larson ; Philippsen, Michael: A DSM protocol aware of both thread migration and memory constraints (Best Paper) . In: Gonzalez, Teofilo F. (Ed.) : Proceedings of the 20th IASTED International Conference on Parallel and Distributed Computing and Systems (20th IASTED International Conference on Parallel and Distributed Computing and Systems Orlando, FL, USA 16.-18.11.2008). Anaheim, CA, USA : ACTA Press, 2008, pp 291-295. - ISBN 978-0-88986-773-4

– Veldema, Ronald ; Philippsen, Michael: Evaluation of RDMA opportunities in an Object-Oriented DSM . In: Adve, Vikram (Ed.) : Languages and Compilers for Parallel Computing (The 20th International Workshop on Languages and Compilers for Parallel Computing (LCPC '07) Illinois 11.-13.10.2007). Vol. LNCS 5234. Berlin : Springer, 2008, pp 217-231. - ISBN 978-3-540-85260-5

– Veldema, Ronald ; Philippsen, Michael: Supporting Huge Address Spaces in a Virtual Machine for Java on a Cluster . In: Adve, Vikram (Ed.) : Languages and Compilers for Parallel Computing (The 20th International Workshop on Languages and Compilers for Parallel Computing (LCPC '07) Illinois 11.-13.10.2007). Vol. LNCS 5234. Berlin : Springer, 2008, pp 187-201. - ISBN 978-3-540-85260-5

– Werth, Tobias ; Dreweke, Alexander ; Wörlein, Marc ; Fischer, Ingrid ; Philippsen, Michael: DAGMA: Mining Directed Acyclic Graphs (Outstanding Paper Award) . In: IADIS (Org.) : Proc. of the 2008 ECDM (IADIS European Conference on Data Mining 2008 Amsterdam, The Netherlands 24.-26.07.2008). Amsterdam, The Netherlands : IADIS PRESS, 2008, pp 11-17. - ISBN 978-972-8924-63-8

– Wilke, Peter ; Ostler, Johannes: Solving the School Time Tabling Problem Using Tabu Search, Simulated Annealing, Genetic and Branch & Bound Algorithms . In: Burke, Edmund K. ; Gendreau, Michel (Ed.) : PATAT '08 Proceedings of the 7th International Conference on the Practice and Theory of Automated Timetabling (7th International Conference on the Practice and Theory of Automated Timetabling (PATAT '08) Montreal, Canada 19.-22.08.2008). 2008, pp 1-4.

1.4 Exam theses (german only)

– Diplomarbeit: Entwurf und Implementierung einer abstrakten Maschine für die oberflächenkompositionale inkrementelle Analyse natürlicher Sprache. Bearbeiter: Johannes Handl (beendet am 14.01.2008); Betreuer: Prof. em. Dr. Hans Jürgen Schneider; Prof. Dr. Michael Philippsen

– Diplomarbeit: Powertype Based Metamodeling als Grundlage für die Modellierung von Software-Entwicklungsprozessen. Bearbeiter: Alexander Hantzsch (beendet am 15.01.2008); Betreuer: Hon.-Prof. Dr.-Ing. Bernd Hindel; Prof. Dr. Michael Philippsen

– Diplomarbeit: Präzisierung und werkzeuggestützte Umsetzung des Software Process Engineering Metamodel 2.0. Bearbeiter: Ralf Ellner (beendet am 16.01.2008); Betreuer: Hon.-Prof. Dr.-Ing. Detlef Kips; Prof. Dr. Michael Philippsen

– Diplomarbeit: Performance- und Laufzeitanalyse eines verteilten Output Management Systems unter Hochlast. Bearbeiter: Christian Hubert (beendet am 01.02.2008); Betreuer: PD Dr.-Ing. habil. Peter Wilke

– Studienarbeit: Analyse von Graph-Daten. Bearbeiter: Sebastian Lenz (beendet am 4.2.2008); Betreuer: Dipl.-Inf. Marc Wörlein; Dipl.-Inf. Tobias Werth; Dipl.-Inf. Alexander Dreweke, B. Sc. mult.; Prof. Dr. Michael Philippsen

– Diplomarbeit: Transformationen von Entwicklungsprozessen in prozessfähige Entwicklungsumgebungen am Beispiel von Microsoft Visual Team System. Bearbeiter: Markus Walter (beendet am 07.03.2008); Betreuer: Hon.-Prof. Dr.-Ing. Bernd Hindel; Prof. Dr. Michael Philippsen

– Diplomarbeit: Analyse und Vergleich von Prozessen und Werkzeugen bei der Entwicklung von HMI-Software für Fahrzeuge und mobile Geräte. Bearbeiter: Matthias Kurz (beendet am 31.3.2008); Betreuer: PD Dr.-Ing. Gabriella Kókai; PD Dr.-Ing. habil. Peter Wilke

– Studienarbeit: Entwurf und Implementierung eines RMI-basierten Controllers für Zeitplanungs-Server-Prozesse. Bearbeiter: Tarek Gasmi (beendet am 15.4.2008); Betreuer: PD Dr.-Ing. habil. Peter Wilke

– Studienarbeit: Entwurf und Realisierung einer Subversion-Schnittstelle für das Prozessmanagementportal project kit. Bearbeiter: Gabriel Dexheimer (beendet am 03.06.2008); Betreuer: Hon.-Prof. Dr.-Ing. Bernd Hindel; Prof. Dr. Michael Philippsen

– Studienarbeit: Implementierung eines LLVM-Backends für Jackal. Bearbeiter: Stefan Kempf (beendet am 26.06.2008); Betreuer: Ronald Veldema, Ph.D.; Prof. Dr. Michael Philippsen

– Master Thesis: Application Migration in Peer-to-peer Compute Clusters. Bearbeiter: Xiaofan Liu (beendet am 5.8.2008); Betreuer: Dipl.-Inf. Michael Klemm; Prof. Dr. Michael Philippsen

– Studienarbeit: Implementierung des JaMP-Programmiermodells für eine Java-VM. Bearbeiter: Georg Dotzler (beendet am 02.10.2008); Betreuer: Dipl.-Inf. Michael Klemm; Prof. Dr. Michael Philippsen

– Diplomarbeit: Entwicklung/Anpassung einer Lattice-Boltzmann-Bibliothek für die CellBE-Architektur. Bearbeiter: Christian Kollee (beendet am 15.10.2008); Betreuer: Dipl.-Inf. Tobias Werth; Prof. Dr. Michael Philippsen

– Studienarbeit: Dynamische Code-Verwaltung für die IBM Cell Broadband Engine. Bearbeiter: Tobias Floßmann (beendet am 31.10.2008); Betreuer: Dipl.-Inf. Dominic Schell; Dipl.-Inf. Michael Klemm; Dipl.-Inf. Tobias Werth; Prof. Dr. Michael Philippsen

– Studienarbeit: Entwurf und Implementierung eines Sicherheitskonzeptes für das Programmierframework FLOW3. Bearbeiter: Andreas Förthner (beendet am 3.12.2008); Betreuer: PD Dr.-Ing. habil. Peter Wilke

– Diplomarbeit: Tracing in UML-Modellen - Nachweis der vollständigen Umsetzung von Anforderungen anhand von Requirements-Tracing in semantischen UML-Modellen. Bearbeiter: Sebastian Lenz (beendet am 12.12.2008); Betreuer: Hon.-Prof. Dr.-Ing. Bernd Hindel; Prof. Dr. Michael Philippsen

– Diplomarbeit: A Review Approach to Software Development Processes using ISO/IEC 15504. Bearbeiter: Philipp Bach (beendet am 16.12.2008); Betreuer: Hon.-Prof. Dr.-Ing. Bernd Hindel

– Diplomarbeit: Design und Implementierung eines regionalen Umfeldmodells zur Erfassung einer Verkehrssituation und exemplarischer Weiterverarbeitung der gewonnenen Daten in Form einer Verkehrsprognose. Bearbeiter: Volker Schmitt (beendet am 23.12.2008); Betreuer: PD Dr.-Ing. habil. Peter Wilke; PD Dr.-Ing. Gabriella Kókai