

## 5 Lehrstuhl für Informatik 2 (Programmiersysteme)

**Anschrift:** Martensstr. 3, 91058 Erlangen

**Tel.:** +49 9131 85 27621

**Fax:** +49 9131 85 28809

**E-Mail:** info@i2.informatik.uni-erlangen.de

### **Leitung:**

Prof. Dr. Michael Philippsen

### **Honorarprofessoren:**

Hon.-Prof. Dr.-Ing. Bernd Hindel

Hon.-Prof. Dr.-Ing. Detlef Kips

### **Emeritus:**

Prof. em. Dr. Hans Jürgen Schneider

### **Sekretariat:**

Agnes Brütting

Waltraud Rück

### **Wiss. Mitarbeiter:**

Dipl.-Inf. Alexander Dreweke, B. Sc. mult.

Dipl.-Ing. (FH) Thorsten Edelhäuser (ab 01.05.2008)

Dipl.-Inf. Ralf Ellner (ab 01.10.2008)

Dipl.-Inf. Philipp Janda

Dipl.-Inf. Michael Klemm (bis 31.08.2008)

Dr.-Ing. Norbert Oster (ab 01.10.2008)

Dipl.-Inf. Dominic Schell (bis 31.03.2008)

Ronald Veldema, Ph.D.

Dipl.-Inf. Tobias Werth

PD Dr.-Ing. habil. Peter Wilke

Dipl.-Inf. Marc Wörlein

### **Gäste/Stipendiaten:**

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

### **Nichtwiss. Personal:**

Dipl.-Ing. (FH) Helmut Allendorf

Manfred Uebler

Zum 1972 gegründeten Lehrstuhl Informatik 2, den Prof. Dr. Michael Philippsen (als Nachfolger von Prof. Dr. H.-J. Schneider) seit April 2002 leitet, gehören neben einem Wissenschaftlerteam, das sich mit **Programmiersystemen** beschäftigt, die **Didaktik der Informatik**, deren Forschungsarbeiten separat dargestellt sind, und die von den Honorar- Professoren geführte Arbeitsgruppe **Praktische Softwaretechnik**.

## **5.1 Forschungsschwerpunkte**

Im Mittelpunkt der **Programmiersystemforschung** des Lehrstuhls stehen parallele und verteilte Systeme und deren Programmierung sowie Programmiersysteme für eingebettete und mobile Systeme. Software (und deren Erstellung) für solche Systeme sollte nicht komplexer, aber genauso portabel, wartbar und robust sein, wie heute für Einprozessorsysteme und Arbeitsplatzrechner. Langfristiges Ziel ist es, den Anwendungen die verfügbare Rechen- und Kommunikationsleistung möglichst ungebremst zur Verfügung zu stellen bzw. aus sehr begrenzten Systemen ein Maximum an Möglichkeiten herauszuholen. Im Bereich der **eingebetteten Systeme** wurde 2008 vor allem an der Minimierung des Speicherbedarfs durch clevere Optimierungen bei der Übersetzung gearbeitet. Im Bereich der **Parallelen Systeme** arbeiten wir weiterhin daran, die Programmierbarkeit solcher Systeme zu verbessern. Dabei beschäftigten wir uns nicht nur mit einzelnen, homogenen Rechnerbündeln sondern auch mit aus heterogenen Einheiten zusammengesetzten Grids. Programmiersysteme für Mehrkernrechner und die damit zusammenhängenden Parallelisierungs- und Optimierungsthemen nahmen 2008 einen immer größeren Raum ein.

Unter der Regie der Honorar-Professoren Dr. Bernd Hindel und Dr. Detlef Kips, die als Geschäftsführer zweier mittelständischer Software-Beratungsunternehmen über langjährige Praxiserfahrung in industriellen Software-Entwicklungsprojekten verfügen, beschäftigt sich die Arbeitsgruppe **Praktische Softwaretechnik** mit wissenschaftlich fundierten Methoden zur standardisierten ingenieurmäßigen Erstellung komplexer Softwaresysteme auf Grundlage wohldefinierter Prozesse. Im Vordergrund stehen dabei insbesondere die systematische Beschreibung, Modellierung und Bewertung von Software-Entwicklungsprozessen sowie deren werkzeuggestützte Anwendung in realistischen Entwicklungsprojekten. Die Forschungsaktivitäten der Arbeitsgruppe konzentrieren sich darauf, geeignete Metamodelle, Notationen und Bewertungsmethoden für realistische Entwicklungsprozesse zu untersuchen und deren Praktikabilität im konkreten Projekteinsatz auf den Prüfstand zu stellen.

## 5.2 Forschungsprojekte

### 5.2.1 Reparallelisierung und Migration von OpenMP-Applikationen

**Projektleitung:**

Prof. Dr. Michael Philippsen

**Beteiligte:**

Dipl.-Inf. Michael Klemm

Ronald Veldema, Ph.D.

**Laufzeit:** 1.11.2003–31.12.2008

**Kontakt:**

Prof. Dr. Michael Philippsen

Tel.: +49 9131 85-27625

Fax: +49 9131 85-28809

E-Mail: philippsen@informatik.uni-erlangen.de

Grid-Computing bietet eine völlig neue Rechnerinfrastruktur. Benutzer erhalten Zugang zu einer Vielzahl von weltweiten Rechenressourcen, die jederzeit über das Internet verfügbar sind. In Zukunft sollen Computational-Grids Anwendern direkten Zugang zu Höchstleistungsrechnern gewähren.

Heutzutage verwenden Benutzer typischerweise einzelne Rechnerbündel eines Computational-Grids und müssen sich mit dem Ablaufplaner befassen, der für die Vergabe von Rechenressourcen verantwortlich ist. Ein Ablaufplaner benötigt die Angabe einer Schätzung der gewünschten Ressourcen, um Teile der Rechanlage exklusiv für eine Applikation zu reservieren. Durch eine gezielt überhöhte Schätzung der benötigten Rechenressourcen versuchen Benutzer, einen vorzeitigen Abbruch der Applikation durch den Ablaufplaner zu vermeiden. Dies verursacht negative Nebeneffekte auf Seiten des Ablaufplaners und verzögert den Start der Applikation des Benutzers.

Dieses Forschungsprojekt wurde mit einer Dissertation erfolgreich abgeschlossen, die eine Lösung präsentiert, welche überhöhte Schätzungen für OpenMP-Applikationen überflüssig macht. Sollten mehr Ressourcen zur Verfügung stehen oder der Applikation entzogen werden, so wird eine Reparallelisierung einer OpenMP-Applikation automatisch durchgeführt. Weiterhin kann eine Applikation zwischen Rechnerbündeln eines Grids migriert werden, sobald die momentan genutzte Ressource kurz vor der erzwungenen Freigabe steht.

Für den Programmierer besteht keine Notwendigkeit, den Quellcode der Applikation zu ändern, um die Funktionen zur Reparallelisierung und Migration zu aktivieren; dies übernimmt der Übersetzer. Die Migration wird mittels eines plattformunabhängigen, koordinierten Sicherungsverfahrens durchgeführt. Eine prototypische Implementierung

eines Migrationsframeworks ermittelt selbständig freie Ressourcen und migriert die Applikation zu diesen Ressourcen. Unsere Messungen zeigen, dass die Laufzeiteinbußen bei geringen 4 % liegen. Diese Einbußen sind angesichts der gewonnenen Flexibilität als akzeptabel anzusehen.

## 5.2.2 Graphbasierte Prozedurale Abstraktion

### **Projektleitung:**

Prof. Dr. Michael Philippsen

### **Beteiligte:**

Dipl.-Inf. Marc Wörlein

Dipl.-Inf. Alexander Dreweke, B. Sc. mult.

Dipl.-Inf. Tobias Werth

**Beginn:** 1.4.2006

### **Kontakt:**

Prof. Dr. Michael Philippsen

Tel.: +49 9131 85-27625

Fax: +49 9131 85-28809

E-Mail: philippsen@informatik.uni-erlangen.de

Als besonders dringend erscheint uns gegenwärtig die Verbesserung der Programmierwerkzeuge für eingebettete Systeme. Solche Systeme werden heutzutage zu oft noch sehr maschinennah programmiert. Das inzwischen bei der Programmierung von Arbeitsplatzrechnern übliche Abstraktions- und Komfortniveau (Objektorientierung, automatische Speicherbereinigung, Ausnahmebehandlung, Parallelität, Aspektorientierung, Komponenten, ...) scheint im Bereich der eingebetteten Systeme noch in weiter Ferne, wodurch Portabilität, Robustheit und Wartbarkeit der erstellten Software erheblich beeinträchtigt wird. Dies ist ein erhebliches volkswirtschaftliches Problem, das gerade auch deshalb bedeutsam ist, weil Europa auf diesem Feld nicht von Amerika dominiert wird. Fernziel muss es daher sein, das Abstraktionsniveau bei der Programmierung eingebetteter Systeme schrittweise zu erhöhen, indem Optimierungstechniken entwickelt werden, die trotz des erhöhten Abstraktionsniveaus "kleinen" Code garantieren.

Neben der offensichtlichen Frage, wie die bekannten Optimierungstechniken auf die Code-Größe wirken, drängen sich neue Einzelfragen auf. Während der RAM-Bedarf einer Applikation auf Desktop-Rechnern kaum eine Rolle spielt, ist dieser für eingebettete Systeme oft essentiell. Objektorientierter - vor allem bibliotheksbasierter - Code bietet ein erhebliches, bislang ungenutztes Potential für prozedurale Abstraktion zur Code-Verkleinerung. Neben der Code-Größe kommt auch dem Aspekt der Energie-Effizienz eine wachsende Bedeutung als Zielgröße der Code-Optimierung zu.

Hier muss der Übersetzer, ggf. im Zusammenspiel mit dem Betriebssystem, optimieren bzw. auf die Hardware-Parameter einwirken. Die Behandlung der nicht-uniformen Speicherzugriffshierarchie, die in verteilten Systemen neben Registern, Cache und Hauptspeicher um eine weitere Leistungsebene vertieft ist, stellt auch bei eingebetteten Systemen eine Herausforderung dar, da z.B. Flash-Speicher zu berücksichtigen sind. Können eingebettete Systeme (ebenso verteilte Systeme) - der Tradition von Desktop-Prozessoren folgend - auch weiterhin mit der Illusion eines transparenten Zugriffs programmiert werden? Kann man durch statische Analyse Informationen über bestehende Lokalitätsbeziehungen zwischen Daten extrahieren? Welche Optimierungen sind dann möglich? Profitieren statische Analyse und Laufzeitmechanismen von einander? Wie können durch Programmanalyse Pre-Fetch- und Post-Store-Kommandos im generierten Code erzeugt werden, durch die Cache-Effekte überdeckt, Wartezeiten vermieden oder Energie gespart werden?

Eine gängige Methode zur Code-Größenverkleinerung ist die Prozedurale Abstraktion (PA): gleiche Code-Abschnitte im Programm werden gesucht und daraus eine einzige neue Prozedur erzeugt. Die Code-Abschnitte werden durch Aufrufe der neu erzeugten Prozedur ersetzt, wodurch die Redundanz innerhalb eines Programms und somit seine Größe reduziert wird. Redundanz entsteht durch die Art und Weise, wie Übersetzer Code generieren (z.B. durch Schablonen). Die bisherigen PA-Ansätze betrachten das Programm als Folge von Instruktionen und suchen nach exakt gleichen Teilfolgen. Sind allerdings Instruktionen innerhalb einer Teilfolge vertauscht, wird sie nicht als weitere Instanz der gesuchten Folge erkannt. Somit fallen potentielle Code-Fragmente für die PA aus der Analyse heraus und das Ergebnis wird suboptimal. Der am Lehrstuhl untersuchte Ansatz löst dieses Problem indem die Instruktionen eines Grundblocks statt in einer Folge in einen Datenflussgraphen (DFG) umgesetzt werden. Ein Graph-Mining-Werkzeug sucht in den DFGs nach gemeinsamen Fragmenten in ARM Assembler-Code, der auf eingebetteten Systemen weit verbreitet ist. In Kooperation mit dem Projekt ParSeMiS, das sich mit der Optimierung von Graph-Minern befasst, werden auch die für PA spezifischen Probleme beim Graph-Minern angegangen. Im Berichtszeitraum wurden die Analysen zur korrekten Rekonstruktion des Datenflussgraphen verfeinert. Eine möglichst genaue Rekonstruktion erhöht das Einsparungspotential im Vergleich zu den herkömmlichen sequentiellen Verfahren. Des Weiteren wurden verschiedene Auslagerungsmethoden optimiert. Diese dienen dazu, die von ParSeMiS als häufig eingestuft Code-Fragmente herauszuziehen und damit das Programm zu verkleinern. Die optimierten Auslagerungsmethoden zeichnen sich vor allem dadurch aus, dass sie möglichst kosteneffizient semantisch gleiche Fragmente vereinheitlichen. Um die Suche nach häufigen Code-Fragmenten zu beschleunigen, wurde 2008 mit der Entwicklung von Heuristiken begonnen.

### 5.2.3 Übersetzerunterstützte Parallelisierung für Mehrkern-Architekturen

**Projektleitung:**

Prof. Dr. Michael Philippsen

**Beteiligte:**

Dipl.-Inf. Tobias Werth

Dipl.-Inf. Dominic Schell

**Beginn:** 1.3.2007

**Kontakt:**

Dipl.-Inf. Tobias Werth

Tel.: +49 9131 85-28865

Fax: +49 9131 85-28809

E-Mail: werth@informatik.uni-erlangen.de

Die Entwicklung von schnelleren und immer effizienteren Rechnerarchitekturen ist in den letzten Jahren an verschiedene Grenzen gestoßen. Althergebrachte Techniken trugen nicht mehr oder nur noch wenig zur Beschleunigung der Hardware bei. Grundprobleme sind dabei das auseinander-driftende Verhältnis der Latenzen von Speicher und CPU und die Abwärme bei steigenden Taktfrequenzen.

Als Lösung drängen sich homogene und heterogene Mehrkern-Architekturen auf, die dem Programmierer enorme Leistung zur Verfügung stellen. Durch verringerte Taktfrequenzen tritt ein Großteil der genannten Problematik nicht auf, die hohe Leistung wird durch Vervielfältigung der Ressourcen erreicht. Somit sind zum Beispiel bei niedrigerem Energieverbrauch mehr Rechenoperationen pro Zeiteinheit möglich. Unter Umständen wird mittels Spezialisierung einzelner Komponenten die Rechenleistung weiter erhöht. Durch eine mehrschichtige Speicherhierarchie mit vielen Zwischenspeichern soll zum Beispiel das Problem der Latenz verkleinert werden.

Aus Mehrkern-Architekturen die volle Leistung abzurufen stellt sich aber als große Schwierigkeit für den Programmierer heraus. Die hohe Rechenkapazität kann er nur dann erreichen, wenn er Expertenwissen sowohl in der Domäne der Anwendung, als auch für die konkrete Architektur besitzt.

Gegenstand der Forschung sind daher unter anderem die folgenden Fragestellungen: Welche Unterstützung kann der Übersetzer dem Programmierer beim Entwickeln von Anwendungen für verschiedenen Mehrkern-Architekturen bieten? Wie viel Kontextwissen ist notwendig, damit der Übersetzer sinnvolle Entscheidungen bei der Parallelisierung auf die einzelnen Kerne trifft? Welchen Anteil der zur Verfügung stehenden Rechenkapazität kann der Programmierer mit vertretbarem Aufwand erreichen, ohne Detailwissen über die Eigenheiten der einzelnen Architekturen besitzen zu müssen? Wie müssen geeignete Werkzeuge zum Auffinden von Fehlern und Flaschenhälsen in der Anwendung auf Mehrkern-Architekturen aussehen?

Ziel dieses Projektes ist es, diese Fragen anhand einer eingeschränkten Anwendungsdomäne zu beantworten und mögliche Lösungswege aufzuzeigen. Als Domäne wird das Lattice-Boltzmann-Verfahren herangezogen, das vor allem in der Strömungssimulation angewandt wird. Durch seine Gitterstruktur und eine überschaubare Anzahl an Datenabhängigkeiten zwischen den einzelnen Zellen lässt sich das Verfahren relativ einfach parallelisieren, so dass sich die Forschung auf die oben genannten Fragestellungen konzentrieren kann.

Im Jahr 2008 wurde in mehrere Richtungen geforscht:

- Es wurden bestehende Programmiermodelle für die CellBE-Architektur verglichen, auf Vor- und Nachteile untersucht und vermessen, um sich ein Bild über den Stand der verwandten Arbeiten zu machen und das Projekt gegenüber anderen Forschungsarbeiten abgrenzen zu können.
- Außerdem wurde für ein bestehendes Programmiermodell (Cilk), das für homogene Mehrkernarchitekturen mit gemeinsamen Speicher gedacht ist, ein Entwurf zur Umsetzung auf der CellBE-Architektur erarbeitet, die heterogen ist und verteilten Speicher besitzt. Dieser Entwurf erweitert die bestehende Sprache um wenige Schlüsselwörter, um die volle Leistung der Architektur parallel nutzen zu können. Dabei werden in einer Quell-zu-Quellcode-Transformation entsprechend gekennzeichnete Regionen auf mehrere Threads für die verschiedenen heterogenen Kerne aufgeteilt. Erste prototypische Auswertungen zeigen zufriedenstellende Resultate.
- Auf den speziellen Kernen der CellBE-Architektur steht nur wenig Speicher (256kB) zur Verfügung. Dieser Platz muss auf Code und Daten aufgeteilt werden. Die gegenwärtig verfolgte Idee beruht darauf, dass nicht der gesamte Programmcode einer Anwendung komplett im Arbeitsspeicher liegen muss, um das Programm auszuführen. Grundlage ist eine am Lehrstuhl erstellte Dissertation (Abschluss der Arbeit im Laufe des Jahres 2009), die analoge Fragestellungen für ARM-Prozessoren mit Scratchpad-Speicher beschreibt. Die am Lehrstuhl entwickelte automatische Speicherverwaltung für Programmcode lädt den Code in Fragmenten (von Basisblöcken bis hin zu Funktionen) in einen Code-Cache. Wenn es zu einem Speicherengpass kommt, wird nicht mehr benötigter Code automatisch erkannt und durch einen speziellen Speicherbereiniger aus dem Speicher entfernt. In Benchmarks zeigt sich, dass der Programmlader gegenüber der nativen Ausführung eine deutliche Einsparung (bis zu 70 %) an Instruktionen bei relativ geringen Laufzeit-Aufwand (plus 5 %) aufweist.

## 5.2.4 Tapir

### **Projektleitung:**

Prof. Dr. Michael Philippsen

### **Beteiligte:**

Ronald Veldema, Ph.D.

Dipl.-Inf. Michael Klemm

**Laufzeit:** 1.1.2006–31.12.2010

### **Kontakt:**

Ronald Veldema, Ph.D.

Tel.: +49 9131 85-27622

Fax: +49 9131 85-28809

E-Mail: veldema@informatik.uni-erlangen.de

Tapir ist eine neue (Programmier-)Sprache zur einfacheren Systemprogrammierung.

Unter Systemprogrammierung versteht man die Programmierung von Netzwerkprotokoll-Software, Betriebssystemen, Middleware, DSM-Systemen usw. Solche Programme sind für das Funktionieren eines Systems essentiell, da sie Systemdienstleistungen bereitstellen, die von Applikationen benutzt werden können. Der Betriebssystemkern stellt einer Applikation z.B. eine Ausführungsumgebung bereit und abstrahiert hierbei von der konkreten Hardware, so dass die Applikation eine rechnerunabhängige Schnittstelle nutzen kann. Ein DSM-System simuliert in einem Rechnerbündel mit verteiltem Speicher einen gemeinsamen Adressraum, damit eine Applikation den gesamten Speicher des Rechnerbündels ohne explizite Kommunikation nutzen kann.

Im Vergleich mit Anwendungssoftware stellt diese Art von Software völlig andere Anforderungen an eine Programmiersprache. Auch unterscheidet sich der angewendete Programmierstil häufig deutlich von Applikationssoftware. Der erzeugte Code muss besonders leistungsfähig sein, da die Leistungsfähigkeit des Systems stark von der Systemsoftware abhängt. Ebenso wirken sich Fehler auf dieser Ebene besonders auf die Zuverlässigkeit der darauf aufbauenden Applikationen aus. Systemsoftware sollte daher (beweisbar) fehlerfrei sein. Diese Anforderungen haben direkte Auswirkungen auf die verwendbaren Programmiersprachen:

- Hochsprachen wie C++, C# und Java verstecken Implementierungsdetails vor dem Programmierer. Der Programmierer benötigt z.B. kein Wissen darüber, wie ein Methodenaufruf konkret durchgeführt wird. Dieses Wissen ist jedoch bei der Entwicklung von Systemsoftware erforderlich.



- Hochsprachen stellen weiterhin Funktionen bereit, die für Systemsoftware in der Regel nicht benötigt werden oder sogar unerwünscht sind. Beispielsweise wird innerhalb eines Betriebssystems explizit keine automatische Speicherbereinigung oder Ausnahmebehandlung verwendet.
- Systemprogramme erfordern kein so hohes Abstraktionsniveau, wie es meist von Hochsprachen gefordert wird. Ebenso verzichtet man bei der Erstellung von Systemsoftware zumeist auf die Benutzung externer Bibliotheken.

Obwohl Tapir an existierende Hochsprachen wie C++ und Java angelehnt ist, wurden alle unnötigen Eigenschaften und Funktionen entfernt. Beispielsweise fehlen Tapir Speicherbereinigung, Ausnahmebehandlung und Typwandlungen; Klassen und Objekte können zwar definiert werden, jedoch ist keine Vererbungsbeziehung zwischen Klassen erlaubt. Das mit Tapir spezifizierte Systemprogramm kann mit Model Checking-Techniken bereits während der Entwicklung auf Fehler überprüft werden. Ein prototypischer Übersetzer und ein Verifikationswerkzeug sind implementiert. Parallel zur Entwicklung der Sprache und der zugehörigen Werkzeuge wird Tapir bereits verwendet, um eine Spezifikation für das DSM-Protokoll von Jackal zu erarbeiten und weitere Arten von DSM-Protokollen zu evaluieren. Eine RDMA-basiertes DSM-Protokoll wurde entwickelt um es in den Tapir-Sprachentwurf einfließen zu lassen. Die semantische Analyse von Tapir-Programmen ist sehr speicherintensiv, da sie auf Modelchecking beruht. Deshalb war es erforderlich, eine eigene Virtuelle Maschine für Java zu konstruieren, die speziell für sehr große Objektmengen ausgelegt ist. Diese neue, LVM genannte virtuelle Maschine zeigt wesentlich bessere Laufzeiteigenschaften als übliche Java-Implementierungen, sobald der verfügbare Hauptspeicher nicht mehr ausreicht und das Auslagern auf den Hintergrundspeicher beginnt.

Im Jahr 2008 lag der Schwerpunkt unserer Arbeiten auf der Weiteentwicklung dieser VM, die nun über mehrere Maschinen verteilt effizient arbeitet. Dadurch können Tapir-Programme schneller verifiziert werden und in Java geschriebene wissenschaftliche Anwendungen laufen schneller ab.

### 5.2.5 JavaParty

**Projektleitung:**

Prof. Dr. Michael Philippsen

**Beteiligte:**

Dipl.-Inf. Marc Wörlein

**Beginn:** 1.4.2007

**Kontakt:**

Prof. Dr. Michael Philippsen

Tel.: +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> erlaubt eine einfache Portierung von parallelen Java-Programmen mit mehreren Threads auf eine verteilte Umgebung wie Cluster. Das Standard-Java unterstützt parallele Programme durch Threads und Synchronisationsmechanismen. Während Mehrprozess-Java-Programme auf einen einzelnen Speicheradressbereich beschränkt sind, dehnt JavaParty die Möglichkeiten Von Java auf verteilte Systeme aus.

Die normale Art parallele Anwendungen auf ein verteilte Umgebung zu portieren ist die Verwendung von Kommunikationsbibliotheken. Java's entfernter Methodenaufruf (RMI) macht die Verwendung expliziter Kommunikationsprotokolle unnötig, aber führt immer noch zu einer erhöhten Programmkomplexität. Der Grund dafür liegt bei den beschränkten Möglichkeiten des RMIs und der benötigten zusätzlichen Funktionalität zur Erzeugung und Dem Zugriff auf entfernte Objekte.

Der Ansatz von JavaParty ist anders. JavaParty Klassen können direkt als entfernt (remote) deklariert werden. Während normale Java Klassen auf eine einzelne Virtuelle Maschine von Java beschränkt ist, sind entfernte Klassen und deren Instanzen in der gesamten verteilten JavaParty Umgebung sichtbar und erreichbar. Soweit man nur entfernte Klassen betrachtet kann die JavaParty Umgebung als ein Virtuelle Maschine angesehen werden, die sich über verschiedene Computer verteilt. Der Zugriff und die Erzeugung von entfernten Klassen ist syntaktisch nicht von dem regulärer Java Klassen zu unterscheiden.

Im Jahr 2008 wurde eine neue Version des JavaParty Übersetzers implementiert, die mit den in Java 1.5/1.6 neu eingeführten Kontrollstrukturen zurecht kommt. Diese Implementierung beruht auf dem öffentlichen und frei verfügbaren Eclipse Übersetzer. Dadurch können zukünftige Weiterentwicklungen der Sprache Java und zugehörige Anpassungen des Übersetzers direkt in JavaParty einfließen.

## **5.2.6 ParSeMiS - die Parallele und Sequenzielle Mining Suite**

### **Projektleitung:**

Prof. Dr. Michael Philippsen

### **Beteiligte:**

Dipl.-Inf. Marc Wörlein

Dipl.-Inf. Alexander Dreweke, B. Sc. mult.

Dipl.-Inf. Tobias Werth

**Beginn:** 1.5.2006

**Kontakt:**

Prof. Dr. Michael Philippsen

Tel.: +49 9131 85-27625

Fax: +49 9131 85-28809

E-Mail: philippsen@informatik.uni-erlangen.de

Die Arbeitsgruppe **ParSeMiS (Parallele und Sequenzielle Graph Mining Suite)** beschäftigt sich mit der Suche nach häufigen interessanten Strukturen in Graphdatenbanken; ein Forschungsgebiet, das in den letzten Jahren sehr reges Interesse geweckt hat. Da viele Forschungs- und Wirtschaftsdaten in strukturierter Form erfasst werden können, bietet sich die Speicherung komplexer Zusammenhänge in Form von allgemeinen oder speziellen Graphen an.

Diese meist unüberschaubaren Datenmengen sind nur schwer mit Hand und Auge zu erfassen, so dass Algorithmen zur Entdeckung interessanter Korrelationen unabdingbar sind. Da deren Entdeckung in Graphen im Allgemeinen aufwändig ist (NP-vollständig), ist die Suche nach parallelen und spezialisierten Algorithmen und Heuristiken notwendig, die den benötigten Rechenzeit- und Speicheranforderungen auch bei immer größer werdenden Datenmengen gewachsen sind.

Das Ziel dieses Projektes ist es, ein effizientes und flexibles Werkzeug zur Suche in beliebigen Graphdaten bereitzustellen, um sowohl die Einbindung in neue Anwendungsgebiete als auch die Entwicklung neuer Suchverfahren zu beschleunigen und zu vereinfachen.

Im Jahr 2008 wurden folgende Ziele erreicht:

- Dokumentation und Veröffentlichung der Quellen zur Verbreitung des Projekts,
- Implementierung einer angepassten graphischen Anzeige für DAGs,
- Beginn der Erneuerung der graphischen Oberfläche und
- Erweiterung der Cluster-Verteilung zur Nutzung aller Kerne bei Bündeln aus Mehrkernrechnern.

### **5.2.7 Modellgetriebene Komponentenkomposition**

**Projektleitung:**

Prof. Dr. Michael Philippsen

**Beteiligte:**

Dipl.-Inf. Philipp Janda

**Laufzeit:** 15.6.2007–14.6.2010

**Förderer:**

AUDI AG

Dieses 2007 im Rahmen der INI.FAU-Kooperation gestartete Projekt soll die Integration von Fahrzeugfunktionen auf Steuergeräte analysieren und modellgetriebene Unterstützungsmöglichkeiten entwickeln. Die gewonnen Erkenntnisse sollen exemplarisch anhand der Integration aller Komponenten eines Fahrdynamikregelsystems auf einem AUTOSAR-Steuergerät überprüft werden.

In der Automobilindustrie ist es schon lange üblich, Fahrzeugfunktionen auf hohem Abstraktionsniveau modellbasiert zu entwickeln. Um frühzeitig Fehleinschätzungen bezüglich Laufzeit- und Ressourcenbedarf auszuschließen, ist es nötig, die entwickelte Software nicht nur zu simulieren sondern auch auf der Zielhardware testen zu können. Aufgrund von Kosten- und Sicherheitsanforderungen ist die Integration auf ein Steuergerät aber sehr zeitaufwändig und erfordert Expertenwissen, das einem Funktionsentwickler normalerweise nicht zur Verfügung steht. AUTOSAR (AUTomotive Open System ARchitecture) scheint sich als Standard für die Basissoftware auf Steuergeräten zu etablieren, doch durch die Neuheit dieses Standards gibt es noch keine Verfahren und Werkzeuge, um die Integration von Funktionen auf einem Steuergerät zu unterstützen.

Im Jahr 2008 wurden die Modellierungsmöglichkeiten in AUTOSAR im Bezug auf ihre Eignung bei Audi und auf mögliche Konflikte mit bestehenden Standards sowie mit bei Audi eingesetzten Technologien untersucht. Des Weiteren wurde die automatische Vervollständigung einer Reglerkomponente zu einer AUTOSAR-Softwarearchitektur prototypisch realisiert. Als zukünftige Unterstützungsmöglichkeiten bei der Integration kommen die automatische Konfiguration der Buskommunikation und das Scheduling der auszuführenden Prozesse in Frage.

### **5.2.8 Integrierte Werkzeug-Kette zur metamodellbasierten Modellierung und Ausführung von Software-Entwicklungsprozessen**

**Projektleitung:**

Hon.-Prof. Dr.-Ing. Detlef Kips

**Beteiligte:**

Dipl.-Inf. Ralf Ellner

Prof. Dr. Michael Philippsen

Dr.-Ing. Martin Jung

Dipl.-Inf. Johannes Drexler

Samir Al-Hilank

**Laufzeit:** 1.10.2008–30.9.2011**Förderer:**

BMW i

**Mitwirkende Institutionen:**

develop group, Erlangen

Aufgrund ständig wachsender Anforderungen, die an die Entwicklung komplexer Softwaresysteme gestellt werden, gewinnt die Einhaltung wohldefinierter Software-Entwicklungsprozesse (SWEPE) immer mehr an Bedeutung. Im Kontext umfangreicher, global verteilter Entwicklungsprojekte ist dabei insbesondere ein Trend zu organisationsübergreifenden, langlaufenden und dabei dynamisch veränderbaren Prozessen erkennbar. Zur effektiven Beschreibung und Unterstützung solcher Entwicklungsprozesse sind speziell geeignete Prozessmodellierungssprachen und eine mächtige Werkzeugunterstützung unverzichtbar.

Im Rahmen einer vom BMW i geförderten Kooperation mit dem Industriepartner develop group wurden im Berichtszeitraum verschiedene existierende Prozessmodellierungssprachen - darunter insbesondere das Software and Systems Process Engineering Metamodel (SPEM) der Object Management Group (OMG) - sowie diverse auf dem Markt verfügbare SWEP-Management-Werkzeuge untersucht und bewertet. Einige wichtige Resultate dieser Untersuchung wurden auf der Konferenz Software Engineering 2008 in München vorgestellt.

Die Ergebnisse der Untersuchung machten deutlich, dass der Markt für SWEP-Beschreibungs- und -Ausführungsumgebungen derzeit noch keine Lösungen bietet, die eine hinreichend präzise und flexible Modellierung von Entwicklungsprozessen sowie deren automatisierte Ausführung, Steuerung und Überwachung ermöglichen. Diese Lücke soll im Rahmen eines weiteren, umfangreicheren Kooperationsprojektes geschlossen werden, das vor kurzem angelaufen ist.

Ziel dieses Projektes ist es, auf Grundlage eines durchgängigen, metamodellbasierten Ansatzes eine integrierte Werkzeugkette für die Modellierung und Ausführung industrieller Software-Entwicklungsprozesse prototypisch zu realisieren. Im Hinblick auf die Praxistauglichkeit der Lösung liegt das Hauptaugenmerk dabei auf der Anpassbarkeit der Prozessmodelle an verschiedene industriellen Entwicklungsszenarien, auf der Anwenderfreundlichkeit der Prozessbeschreibung und auf einer weitgehenden Automatisierung der Prozessausführung, die zur Effizienzsteigerung in der Entwicklung entscheidend beiträgt. Diese charakteristischen Vorzüge sollen durch einen relativ hohen Formalisierungsgrad der Prozessmodellierung, durch eine weitgehende Generizität der Modellierungs- und Prozessausführungswerkzeuge sowie durch die Verwendung verbreiteter und akzeptierter Industriestandards (UML, SPEM) erreicht werden.

Dieses Projekt wird ebenfalls in Zusammenarbeit mit der develop group als Industriepartner durchgeführt und mit Mitteln des BMW i gefördert. Es wurde im Oktober 2008 mit drei wissenschaftlichen Mitarbeitern gestartet und ist auf insgesamt drei Jahre aus-

gelegt.

### **5.2.9 Funkortung von Antennenpositionen**

**Projektleitung:**

PD Dr.-Ing. Gabriella Kókai

**Beteiligte:**

Dipl.-Ing. (FH) Thorsten Edelhäuser

**Laufzeit:** 1.5.2008–30.4.2011

**Förderer:**

Fraunhofer Institut für Integrierte Schaltungen

Im Jahr 2008 wurde eine Software entwickelt, die es ermöglicht, die Position der Empfangsantenne eines Ortungssystems zu ermitteln. Dabei wurden Roboter benutzt, um die Position und die Ausrichtung der Antenne festzustellen. Der Roboter ermittelt an verschiedenen Positionen die charakteristischen Merkmale des Ortungssystems und speichert diese Daten ab. Unser entwickeltes Verfahren nutzt diese Daten, um die Position und Ausrichtung der Antenne mithilfe von heuristischen Optimierungsalgorithmen zu bestimmen. Versuche in realen Experimenten bestätigen die Anwendbarkeit unseres Verfahrens in der Praxis.

### **5.2.10 Evolutionäre Agenten**

**Projektleitung:**

Dipl.-Inf. Stephan Otto

**Beginn:** 1.4.2008

**Kontakt:**

Dipl.-Inf. Stephan Otto

Tel.: +49 9131 85-27830

Fax: +49 9131 85-28809

E-Mail: Stephan.Otto@informatik.uni-erlangen.de

Die starke Vernetzung von Computersystemen hat in den letzten Jahren zu enormen Veränderungen in nahezu allen Bereichen geführt. Beispiele hierfür sind das Internet, Grid-Computing, Peer-to-Peer Netzwerke und darauf aufbauende Verfahren wie z.B. Agentensysteme. Diese Entwicklung ist partiell das Resultat verteilter Systeme und den damit entstandenen inhärent verteilten Problemen. Eine Konsequenz ist zunehmende Dezentralität beim Lösen verteilter Probleme, da zentrale Verfahren unzureichend dafür geeignet sind:

- Zentrale Ressourcen sind begrenzt in ihrer Fähigkeit, (alle notwendigen) Daten zu speichern, zu übertragen und zu verarbeiten,
- In unternehmensübergreifenden Geschäftsumgebungen existiert eine kommunikationseinschränkende Informationsasymmetrie,
- Dynamik: während zentral eine Lösung erstellt wird, hat sich das Problem bereits verändert.

In diesem Umfeld dynamischer, nicht zugreifbarer und verteilten Strukturen stellt die Adaption und Optimierung von Systemen und Geschäftsprozessen ein nach wie vor nur unzureichend gelöstes Problem dar. Im Allgemeinen sind Adaptions- bzw. Optimierungsverfahren so ausgelegt, dass Informationen für inhärent verteilte Probleme zentral gesammelt und bearbeitet werden, um ein möglichst gutes Ergebnis zu erzielen. Es existieren eine Reihe von speziellen Ansätzen zur verteilten Optimierung, deren Funktionsweise auf ein Problem bzw. eine eingeschränkte Anzahl von Problemen zugeschnitten ist. Häufig müssen die Akteure kooperativ zusammenarbeiten, um eine Lösung zu erreichen. Bei vorhandenen Optimierungsverfahren finden sich wesentliche, noch nicht ausreichend untersuchte Problembereiche der Verteilung (Daten, Ressourcen), Heterogenität und dynamische Umwelt. Hierbei liegt der Fokus insbesondere auf der Toleranz gegenüber dynamischen, verteilten und heterogenen Basisressourcen.

Dieses Projekt fokussiert ein gegenüber den genannten Problembereichen tolerantes Optimierungsverfahren. Während sogenannte Top-down Verfahren ausgehend von einem zentralen Ansatz arbeiten, wird in diesem Projekt ein sogenannter Bottom-up Ansatz verfolgt. Ebenso wie in natürlichen komplexen Systemen entstehen komplexe Softwaresysteme aus dem Zusammenspiel sogenannter Agenten, indem jeder Agent einfachen lokalen Verhaltensmustern folgt. Selbstorganisation und damit verbunden Adaption ist ein Hauptmerkmal komplexer Systeme. Im Rahmen dieses Projektes wird ein neues generisches Konzept verteilter Optimierung mittels evolutionärer Agenten verfolgt. Es werden dezentrale Operatoren für die Selektion und Rekombination verwendet, die auf ökonomischen Marktmechanismen basieren. Damit kann der Flaschenhals zentraler Selektion aufgrund berechneter Fitnesswerte umgangen werden. Ein dezentrales bottom-up Adaptions- und Optimierungsverfahren kann somit erforscht und in unterschiedlichen Szenarien erprobt werden. Die Methode basiert auf einem formalen Modell, welches den Adaptionsmechanismus für die Anzahl und Strategie der einzelnen Agenten erklärt und damit die entstehende emergente Optimierung offenlegt.

Der Beitrag dieses Projektes ist im Wesentlichen wie folgt zusammenzufassen:

- Entwicklung eines neuartigen verteilten Evolutionären Algorithmus, um den üblicherweise zentral ablaufende Fitnessvergleich und die zentrale Selektion zu vermeiden

- Entwicklung von endogener Fitness und ihrer Auswirkung auf die Ergebnisqualität
- Es wurden neue lokale Selektionsverfahren entwickelt und miteinander verglichen.
- Um die grundlegende Fähigkeit des Verfahrens zu zeigen, wurden empirische Studien zur Takeover-time durchgeführt. Hiermit wurde gezeigt dass der Selektionsdruck vergleichbar mit klassischen evolutionären Verfahren ist.
- Ein neues Dezentralitätsmaß wurde entwickelt, um verteilte Ansätze hinsichtlich ihres Grades an Dezentralität einordnen zu können.
- Das entwickelte Konzept wurde an unterschiedliche Anwendungsfälle adaptiert und damit seine Leistungsfähigkeit unter Beweis gestellt.

### 5.2.11 Optimierung von FIR-Filterstrukturen

**Projektleitung:**

PD Dr.-Ing. Gabriella Kókai

**Beteiligte:**

Dipl.-Inf. Szilvia Zvada

Dipl.-Ing. Hans Holm Frühauf

**Laufzeit:** 1.1.2006–30.9.2009

**Kontakt:**

PD Dr.-Ing. Gabriella Kókai

Tel.: +49 9131 85-28996

Fax: +49 9131 85-28809

E-Mail: kokai@informatik.uni-erlangen.de

Dank der rapiden Verbreitung elektronischer Systeme im Alltag rückten VLSI-Chips (very large scale integration) schnell in den Fokus der Forschung. Das Hauptziel in diesem Bereich ist der Entwurf von kleinen und schnellen Chips bei gleichzeitig niedrigem Energieverbrauch. Der Fortschritt der modernen Chipherstellungstechnologie und die zunehmende Packungsdichte haben es ermöglicht, dass heutige VLSI-Chips einige Millionen Transistoren enthalten. Aus der Sicht eines Chipdesigners bedeutet dieses eine beträchtliche Zahl potenzieller Chipstrukturen bei der Suche nach einem optimalen oder annähernd optimalen Chip. So wird die Automatisierung des Designprozesses in zunehmendem Maße wichtig.

Im Fall digitaler Filter liegt die Aufmerksamkeit vor allem auf dem Design von FIR-Filtern (finite impulse response). Diese Filter werden allgemein verwendet, um digitale



Datenströme gemäß einer linearen Funktion umzuwandeln, wie z.B. bei der Linearisierung durch Endverstärker oder bei der Kalibrierung von Audio- oder Videoempfängern. Wenn jedoch die Aufgabe eine nicht-lineare Transformation der Datenströme ist, muss ein manueller und daher zeitraubender Entwurf solcher Filter durchgeführt werden.

Das in diesem Projekt entwickelte evolFIR System schließt diese Lücke, indem es ein neuartiges Entwurfswerkzeug zur Verfügung stellt, das das Logikdesign der polynomischen FIR-Filter-Strukturen optimieren kann. Auf dieser Ebene des Chipdesigns sind Funktionselemente wie Addierer oder Multiplizierer und logischen Primitive wie Verschiebe- und Verzögerungselemente miteinander kombiniert, um die benötigte Funktionalität der gewünschte Filter zu gewährleisten. Diese Funktionalität ist durch die Übertragungsfunktion des Filters definiert. Während des evolutionären Prozesses müssen wir einerseits sicher stellen, dass die Individuen stets exakt die vorgegebene Übertragungsfunktion beschreiben. Andererseits müssen die betrachteten Topologien bestimmte hardware-spezifische Anforderungen erfüllen, wie z.B. die begrenzte Anzahl an Eingängen der jeweiligen Blockelemente.

Die zentrale Aufgabe für den Evolutionsprozess in evolFIR ist es, eine kleine (möglichst wenige Blockelemente enthaltende), redundanzfreie Filterstruktur zu finden. Dies erreichen wir durch die Anwendung des AGGP-Verfahrens (Genetische Programmierung basierend auf attributierten Grammatiken) auf folgenden Weise:

- Die Individuen des evolutionären Prozesses sind spezielle Ableitungsbäume, die die mögliche Topologie der Funktionselemente und der logischen Primitive darstellen. Da evolFIR die Anwendung von Multiplizierern in Filterkompositionen unterstützt, können auch polynomielle Übertragungsfunktionen optimiert werden.
- Mithilfe der Attribute und des speziellen Zufallsbaumgenerators von AGGP ist sichergestellt, dass ausschließlich solche Ableitungsbäume während der Optimierung erzeugt werden, die genau die Ziel-Übertragungsfunktion repräsentieren.
- Dieser spezielle Zufallsbaumgenerator berücksichtigt außerdem auch variierende Einschränkungen, die für den späteren Hardware-Synthese-Prozess relevant sind.
- Durch die Verwendung einer speziellen Darstellungsform (abstrakt-verkettete Ableitungsbäume), werden redundante Teile der erzeugten Filterstrukturen nicht nur reduziert, sondern komplett eliminiert.

Die obigen Einschränkungen und unsere besondere Darstellungsform zusammen führen dazu, dass der Lösungsraum des evolFIRs inhomogen und die Fitnessfunktion diskontinuierlich ist. Aus diesem Grund erfordert die Parametrisierung des evolutionären Kerns des evolFIRs besondere Aufmerksamkeit, um eine verfrühte Konvergenz des evolutionären Prozesses vorzubeugen. Im Jahr 2008, haben wir eingehend die Auswirkungen

der hardware-spezifischen Einschränkungen und der Parameter auf den Lösungsraum bzw. deren Wechselwirkung mit den evolutionären Parametern untersucht.

### **5.2.12 Zeitplanungsalgorithmen**

**Projektleitung:**

PD Dr.-Ing. habil. Peter Wilke

**Beteiligte:**

Dipl.-Inf. Johannes Ostler

**Laufzeit:** 1.1.2004–31.12.2010

**Kontakt:**

PD Dr.-Ing. habil. Peter Wilke

Tel.: +49 9131 85-27624

Fax: +49 9131 85-28809

E-Mail: wilke@informatik.uni-erlangen.de

Zeitpläne müssen in vielen unterschiedlichen Bereichen erstellt werden, z.B. in der Schulstundenplanung oder der Personaleinsatzplanung. Da es sehr mühsam ist, komplexe Zeitpläne wie Schulstundenpläne per Hand zu erstellen, werden die meisten Zeitpläne computerunterstützt generiert. Dazu wurde am Lehrstuhl in den vergangenen Jahren eine Software entwickelt, die es ermöglicht, die Planung unter zu Hilfenahme verschiedener Optimierungsalgorithmen durchzuführen. Diese Version der Zeitplanungssoftware wurde aus einer auf genetischen Algorithmen basierenden Version weiterentwickelt, wobei sich zeigte, dass einige Erweiterungen wegen der notwendigen Kompatibilität zur Grundversion nicht optimal implementieren ließen.

**Erlangen Advanced Time Tabling Software EATTS** ist die innovative Entwicklungs- und Produktionsumgebung zur Erstellung optimierter Zeitplanungen.

**Ressourcen**

Zeitplanungsprobleme treten in der Praxis in verschiedenen Formen auf: Schichtpläne, Fertigungspläne, Stundenpläne u.v.a. Allen gemeinsam ist, dass bestimmte Ereignisse unter Berücksichtigung von Randbedingungen möglichst optimal geplant werden müssen. Das Ergebnis der Planung ist dann ein Zeitplan. Im Beispiel der Schulplanerstellung wären die Ereignisse Schulstunden, denen Ressourcen wie Lehrer, Klassen und Räume zugeordnet werden müssen. Die Ressourcen werden in Typen unterteilt. Für jeden dieser Typen können beliebig viele Attribute vom Benutzer definiert werden.

Eine Zeitplanerstellung beginnt typischerweise mit der Erfassung der einzuplanenden Ressourcen. Diese kann durch Import eines Datenbestandes oder manuelle Erfassung geschehen.

## **Ergebnisse**

Als Ergebnisse der Planungsalgorithmen werden Zeitpläne erstellt. Diese können in verschiedenen Formaten gespeichert und angezeigt werden. So ist es z. B. möglich, verschiedene Sichten auf einen Plan zu erzeugen.

Typisch ist die Anbindung über einen Browser, d.h. den einzelnen Benutzern werden entsprechend ihren Privilegien die Sichten und Funktionen zur Verfügung gestellt.

## **Randbedingungen**

Die Beschreibung von Randbedingungen ist meist viel komplexer als die von Ressourcen und Ereignissen.

Zum Einen müssen die Randbedingungen exakt formuliert werden, zum Anderen darf die Übersichtlichkeit nicht verloren gehen, um z. B. Widersprüche oder Lücken entdecken zu können, die ja leider nicht automatisch gefunden werden können. Randbedingungen kommen in vielen Varianten vor, weshalb eine flexible Spezifikation notwendig ist. In der Spezifikation kann auf Ressourcen und/oder deren Attribute, die ja vom Benutzer definiert werden, zugegriffen werden. Abhängig vom Typ dieser Variablen, unter anderem Integer, Gleitkomma und Zeichenketten, stehen Verknüpfungs- und Vergleichsoperatoren zur Verfügung, um die Bedingungen zu formulieren. Zusätzlich werden die Parameter der Kostenfunktion gewählt, um bei einer Verletzung der Randbedingung die entsprechenden Strafpunkte zu berechnen.

Eine Besonderheit unserer Software ist, dass Randbedingungen nicht nur als "unbedingt einzuhalten (hard)" oder "nach Möglichkeit einzuhalten (soft)" klassifiziert werden können, sondern auch als "darf im Ausnahmefall verletzt werden (soft hard)". Somit kann die Verletzung bestimmter Randbedingungen im Ausnahmefall erlaubt werden. So kann beispielsweise flexibel auf den Ausfall von Ressourcen reagiert werden, indem ein neuer Zeitplan erstellt wird, der möglichst wenig Abweichungen vom bisherigen Plan hat, z. B. muss ja nicht der gesamte Stundenplan aller Schüler neu erstellt werden, nur weil ein Lehrer krank geworden ist, oder ein Klassenraum wegen eines Rohrbruchs nicht benutzbar ist. In diesen Fällen soll nur ein Vertretungsplan erstellt werden,

## **Algorithmen**

Herzstück der Planung sind die verwendeten Algorithmen. Abhängig von der Natur der Randbedingungen und den gewünschten Eigenschaften kann aus einer Vielzahl von bereits implementierten Algorithmen ausgewählt werden: Genetische Algorithmen - Evolutionäre Algorithmen - Branch-and-Bound - Tabu Search - Simulated Annealing - Graphenfärbung - Soft Computing - Schwarm Intelligenz

Für den Einstieg stehen vorkonfigurierte Algorithmen zur Verfügung, der fortgeschrittene Benutzer kann aber die Parameter der Algorithmen an seine Bedürfnisse anpassen

oder neue Algorithmen implementieren. Alle diese Algorithmen können in Experimenten beliebig zu Berechnungssequenzen kombiniert werden. Die Konfiguration eines Experiments kann abgespeichert werden und z. B. als Vorlage für ein neues Experiment dienen oder nochmals ausgeführt werden.

### **Ausführung von Experimenten**

Die Algorithmen werden entweder auf einem dedizierten Server ausgeführt und bei Bedarf über das TCP/IP-Protokoll auf weitere Rechner verteilt. Die Abbildung zeigt den Dialog zur Auswahl und zum Start der Experimente und die Übersicht der laufenden Experimente. Der Browser verbindet sich in regelmäßigen Abständen automatisch mit dem Server und erhält von diesem den aktuellen Stand der Berechnung. Dieser Statusinformationen beinhalten unter anderem die Kosten des bisher besten gefundenen Plans sowie eine Abschätzung für die verbleibende Berechnungszeit. Nach Beendigung der Berechnung werden die Ergebnisse gespeichert und die Dateien, die zur Visualisierung der Pläne nötig sind erstellt. Der Planer kann nun entscheiden, ob die Qualität der gefundenen Lösung ausreichend ist, oder ob er auf ihrer Basis weitere Optimierungsläufe starten will.

### **Ergebnisse**

Als Ergebnisse der Planungsalgorithmen werden Zeitpläne erstellt. Diese können in verschiedenen Formaten gespeichert und angezeigt werden. So ist es z.B. möglich verschiedene Sichten auf den Plan zu erzeugen.

Typisch ist die Anbindung über einen Browser, d.h. den einzelnen Benutzern werden entsprechend ihren Privilegien die Sichten und Funktionen zur Verfügung gestellt.

### **Zusammenfassung**

Die Software ist in Java implementiert und damit plattform-übergreifend verfügbar, insbesondere für die Betriebssysteme Windows und Linux.

Für den Betrieb von EATTS werden folgende frei verfügbare kostenlose Software-Produkte benötigt:

- ein JavaScript-fähiger Browser zur Anzeige der Bedienoberfläche

Optional kann ein dedizierter EATTS-Server konfiguriert werden. Dazu wird benötigt:

- Java Laufzeitumgebung (JRE Java Runtime Environment) (min v5.0)
- über TCP/IP Netzwerk erreichbare Rechner zur verteilten Berechnung (optional)

**2008**

Im Jahr 2008 wurde die Struktur der Algorithmen optimiert um die nebenläufige Berechnung zu beschleunigen. Dies soll in Zukunft auf Rechner mit Multi-Core-Prozessoren ausgedehnt werden.

Da es sich die Installation der Software durch die potentiellen Nutzer als zu komplex herausgestellt hat, wurde eine abgespeckte Version implementiert, die keine Datenbank mehr benötigt, sondern deren Datenhaltung und Austausch auf XML-Dokumenten basiert. Zusätzlich wird eine Variante angeboten, bei der die Nutzer ihre Experimente auf einem an der Universität Erlangen installierten Server rechnen lassen können.

Die Oberfläche der Software wurde komplett als web-basierte Anwendung reimplementiert.

Auf der CeBIT 2009 wird die neue Version der Software, die jetzt EATTS Erlangen Advanced Time tabling System heisst, vorgestellt werden.

### **5.2.13 Graphen und Graphtransformationen**

#### **Projektleitung:**

Prof. em. Dr. Hans Jürgen Schneider

**Laufzeit:** 1.10.2004–30.9.2010

#### **Kontakt:**

Prof. em. Dr. Hans Jürgen Schneider

Tel.: +49 9131 85-27620

Fax: +49 9131 85-28809

E-Mail: [schneider@informatik.uni-erlangen.de](mailto:schneider@informatik.uni-erlangen.de)

Graphen werden an vielen Stellen als intuitives Hilfsmittel zur Verdeutlichung komplizierter Sachverhalte verwendet. Außerhalb der Informatik trifft dies z.B. auf die Biologie oder Chemie zu, wo Moleküle graphisch modelliert werden. Innerhalb der Informatik werden Daten- bzw. Kontrollflussdiagramme, Entity-Relationship-Diagramme oder Petri-Netze zur Visualisierung sowohl von Software- als auch von Hardware-Architekturen häufig verwendet. Graphgrammatiken und Graphtransformationen kombinieren Ideen aus den Bereichen Graphentheorie, Algebra, Logik und Kategorientheorie, um Veränderungen an Graphen formal zu beschreiben.

Die zugrundeliegende Theorie ist ein attraktives Hilfsmittel, äußerst unterschiedliche Strukturen in einer einheitlichen Weise zu beschreiben, z.B. die unterschiedlichen Modelle für asynchrone Prozesse: Petri-Netze basieren auf gewöhnlichen markierten Graphen, Statecharts verwenden hierarchische Graphen, die parallele logische Programmierung kann mit Hilfe sogenannter Dschungel graphentheoretisch interpretiert werden, und die Aktorsysteme lassen sich als Graphen darstellen, deren Markierungsalphabet eine Menge von Termgraphen ist.

Im Jahre 2008 haben wir sowohl ein theoretisches Konzept untersucht als auch einen Implementierungsaspekt betrachtet:

- Eine Datenbanktransaktion wird durch eine Sequenz atomarer Transformationen beschrieben, die aber von außen als unteilbarer Schritt angesehen wird. Aus der Sicht der Graphtransformationssysteme lautet die Frage: Können wir, wenn eine Ableitungssequenz gegeben ist, eine Produktion konstruieren, die die Wirkung der gesamten Ableitungssequenz in einem Ableitungsschritt simuliert? Die Fragestellung wurde bereits in unserem Einführungsaufsatz (1973) untersucht, aber beschränkt auf injektive Graphmorphismen. H. Ehrig und H.-J. Kreowski haben das Ergebnis 1976 auf markierte Graphen übertragen, und H. Ehrig erlaubte 1977 nichtinjektive Einbettungen, aber seine Darstellung verwendet mengentheoretische Argumente und macht keinen Gebrauch von den kategoriellen Konstruktionen. Im Jahr 2006 haben Ehrig et al. eine rein kategorielle Darstellung vorgelegt, die aber nur den Spezialfall der adhäsiven Kategorien betrachtet und sich ebenfalls auf Monomorphismen beschränkt. Jetzt konnten wir zeigen, dass diese restriktiven Einschränkungen unnötig sind. Unsere Lösung ist ebenfalls rein kategoriell, macht aber keinerlei Einschränkungen bei den Morphismen. Darüber hinaus verlangen wir nur, dass die betrachtete Kategorie Pushouts und Pullbacks besitzt; das bedeutet, dass wir das Ergebnis auf jede praktisch interessante Kategorie anwenden können, insbesondere auf die Kategorie der strukturiert markierten Graphen, die nicht adhäsiv ist. Der Beweis ist im Entwurf zu unserem Lehrbuch enthalten: <http://www2.informatik.uni-erlangen.de/Personen/schneide/gtbook/chapter5.pdf>
- Die kategorielle Behandlung der Graphtransformationen ist hochgradig generisch: Alle Beweise und Konstruktionen gelten für verschiedene Graphtypen. Nur die grundlegenden Operationen müssen für jede Anwendung detailliert beschrieben werden, die darauf aufbauenden kategoriellen Eigenschaften sind dann automatisch definiert. Da moderne Programmiersprachen generische Konzepte unterstützen, sieht es vielversprechend aus, den kategoriellen Ansatz zur Beschreibung von Graphtransformationssystemen in Sprachen wie Java oder Haskell zu implementieren. Java benutzt Klassen von Objekten, aber es unterstützt die Mehrfachvererbung nicht wirklich, da Schnittstellen (Interfaces) keine Methodendefinitionen enthalten dürfen. Deshalb müssen eigene Konstruktionsklassen (factory classes) eingeführt werden, die die generischen kategoriellen Konstruktionen implementieren und die von jeder implementierten Kategorie importiert werden müssen. Dagegen unterstützt Haskell die Mehrfachvererbung, betrachtet aber Klassen von Typen und verlangt, dass konkrete Typen explizit zu Instanzen aller Klassen gemacht werden müssen, zu denen sie gehören. Unsere Pilotimplementierungen werfen interessante Fragen bezüglich der unterschiedlichen Sicht von Generi-

zität auf. Die Haskell-Version ist bereits verfügbar: <http://www2.informatik.uni-erlangen.de/Personen/schneide/gtbook/appendix-a.pdf> Die Java-Version wird demnächst verfügbar sein. Die wesentlichen Teile sind jedoch in dem Begleitmaterial zur Vorlesung über Graphtransformationssysteme zu finden: <http://www2.informatik.uni-erlangen.de/Lehre/WS200809/GraTra/index.xml>

#### **5.2.14 International Collegiate Programming Contest an der FAU**

**Projektleitung:**

Prof. Dr. Michael Philippsen

**Beteiligte:**

Dipl.-Inf. Tobias Werth

Dipl.-Inf. Marc Wörlein

Dipl.-Inf. Alexander Dreweke, B. Sc. mult.

**Beginn:** 1.11.2002

**Kontakt:**

Dipl.-Inf. Tobias Werth

Tel.: +49 9131 85-28865

Fax: +49 9131 85-28809

E-Mail: [werth@informatik.uni-erlangen.de](mailto:werth@informatik.uni-erlangen.de)

Die Association for Computing Machinery (ACM) richtet seit Jahrzehnten den International Collegiate Programming Contest (ICPC) aus. Dabei sollen Teams aus je drei Studenten in fünf Stunden neun bis zehn Programmieraufgaben lösen. Als Erschwernis kommt hinzu, dass nur ein Computer pro Gruppe zur Verfügung steht. Die Aufgaben erfordern solide Kenntnisse von Algorithmen aus allen Gebieten der Informatik und Mathematik, wie z.B. Graphen, Kombinatorik, Zeichenketten, Algebra und Geometrie.

Der ICPC wird jedes Jahr in drei Stufen ausgetragen. Zuerst werden innerhalb der Universitäten in lokalen Ausscheidungen die maximal drei Teams bestimmt, die dann zu den regionalen Wettbewerben entsandt werden. Erlangen liegt seit dem Jahr 2009 im Einzugsbereich des Northwestern European Regional Contest ([NWERC]<http://2009.nwerc.eu>), an dem u.a. auch Teams aus der Großbritannien, den Benelux-Staaten und Skandinavien teilnehmen. Die Sieger aller regionalen Wettbewerbe der Welt (und einige Zweitplatzierte) erreichen die World Finals, die im Frühjahr des jeweils darauffolgenden Jahres (2009 in Stockholm) stattfinden.

Im Jahr 2008 fanden zwei lokale Wettbewerbe an der FAU statt. Im Wintersemester wurde ein Mannschaftswettbewerb ausgetragen mit dem Ziel, neue Studierende für die Wettbewerbe zu begeistern mit einer Rekordzahl von 18 Teams. Jedes Team bestand

aus maximal drei Studenten. Außerdem nahmen noch Teams der TU München sowie der Universität Konstanz online am Wettbewerb teil.

Im Sommersemester fand zum wiederholten Mal das Hauptseminar "Hallo Welt! - Programmieren für Fortgeschrittene" statt, um Studierende verschiedener Fachrichtungen in Algorithmen und Wettbewerbs-Aufgaben zu schulen. Der Wettbewerb im Sommersemester diente der Auswahl der studentischen Vertreter der FAU für den SWERC 2008, der dieses Jahr von der FAU in Nürnberg ausgerichtet wurde. Insgesamt nahmen am lokalen Ausscheidungskampf 25 Studierende der verschiedensten Fachrichtungen teil. Die besten neun bildeten Dreier-Teams (der Zehntplatzierte wurde als Ersatzmann ausgewählt) und errangen beim südwesteuropäischen Wettbewerb gegen die internationale Konkurrenz die Plätze 4, 10 und 26 von insgesamt 56 teilnehmenden Teams. Das beste Team holte sich somit eine Goldmedaille, das zweitbeste Team eine Silbermedaille. Auch 2008 zeigte das Trainingslager somit den gewünschten Erfolg.

Der [Regionalausscheid]<http://icpc.informatik.uni-erlangen.de/swerc2008/> war ein besonderer Erfolg, da er erstmals von der FAU ausgerichtet wurde. Dabei kamen knapp 60 Teams verschiedener Universitäten nach Nürnberg, um die Programmierkrone für Südwesteuropa zu erkämpfen. Gewonnen hat ein Team der Ecole Normale Supérieure aus Lyon. Auch im kommenden Jahr wird die FAU den Regionalausscheid ausrichten, diese Mal für Nordwesteuropa.

### 5.3 Publikationen

- 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 (Hrsg.) : EuroPar 2008 - Parallel Processing (Proceedings of the Euro-Par 2008 Conference Las Palmas de Gran Canaria, Spain 26.-29.08.2008). Bd. LNCS 5168. Heidelberg, Germany : Springer, 2008, S. 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 ; Brüggel, Bernd (Hrsg.) : Software Engineering 2008: Fachtagung des GI-Fachbereichs Softwaretechnik (Software Engineering 2008 München 18.-22.2.2008). Bd. 121. LNI : GI, 2008, S. 67-74. - ISBN 978-3-88579-215-4
- Dreweke, Alexander: Graph-Based Procedural Abstraction . Saarbrücken : VDM, 2008. - 60 Seiten. ISBN 978-3-8364-6568-7
- Dreweke, Alexander: Lattice Boltzmann Method for DSM Systems . Saarbrücken : VDM, 2008. - 56 Seiten. 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. - Abschlußbericht (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 (Hrsg.) : 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, S. 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. (Hrsg.) : 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, S. 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), S. 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 (Hrsg.) : Applications of Evolutionary Computing - EvoWorkshops 2008: EvoCOMNET, EvoFIN, EvoHOT, EvoIASP EvoMUS-ART, 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, S. 659-668. (Lecture Notes in Computer Science Bd. 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 (Hrsg.) : 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, S. 415-422. (Lecture Notes in Informatics Bd. 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 (Hrsg.) : INFORMATIK 2008 - Beherrschbare Systeme dank Informatik (Band 1) (Workshop Modellbasiertes Testen München 09.09.2008). Bd. 1. Bonn : Gesellschaft für Informatik, 2008, S. 193-198. (Lecture Notes in Informatics Bd. 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 (Hrsg.) : 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, S. 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), Nr. 6, S. 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), Nr. 1, S. 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. (Hrsg.) : 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, S. 291-295. - ISBN 978-0-88986-773-4
- Veldema, Ronald ; Philippsen, Michael: Evaluation of RDMA opportunities in an Object-Oriented DSM . In: Adve, Vikram (Hrsg.) : Languages and Compilers for Parallel Computing (The 20th International Workshop on Languages and Compilers for Parallel Computing (LCPC '07) Illinois 11.-13.10.2007). Bd. LNCS 5234. Berlin : Springer, 2008, S. 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 (Hrsg.) : Languages and Compilers for Parallel Computing (The 20th International Workshop on Languages and Compilers for Parallel Computing (LCPC '07) Illinois 11.-13.10.2007). Bd. LNCS 5234. Berlin : Springer, 2008, S. 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 (Veranst.) : 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, S. 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 (Hrsg.) : 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, S. 1-4.

## 5.4 Studien- und Abschlussarbeiten

- 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