

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

Ordinarius:

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

Margit Zenk

Techniker/Administatoren:

Dipl.-Inf. (FH) Helmut Allendorf

Manfred Übler

Wiss. Mitarbeiter:

Dipl.-Inf. Thorsten Blaß

Dipl.-Inf. Daniel Brinkers (ab 10.01.2011)

Dipl.-Inf. Georg Dotzler

Dr. Alexander Dreweke (bis 31.01.2011)

Dr. Thorsten Edelhäuser (bis 30.09.2011)

Dipl.-Inf. Ralf Ellner (bis 30.09.2011)

Dipl.-Inf. Philipp Janda (bis 31.01.2011)

Dipl.-Inf. Stefan Kempf

Dipl.-Math. Jakob Krainz (ab 01.07.2011)

M. Sc. Andreas Kumlehn (ab 15.09.2011)

Dipl.-Inf. Christopher Mutschler

Dr.-Ing. Norbert Oster

M. Eng. Norbert Tausch, Dipl.-Ing. (FH)

PD Ronald Veldema, Ph.D.

Dipl.-Inf. Tobias Werth

Dipl.-Inf. Marc Wörlein (bis 31.01.2011)

Gäste:

Dipl.-Inf. (FH) Josef Adersberger

Sören Stange, M.Sc.

Externes Lehrpersonal:

Dr.-Ing. Klaudia Dussa-Zieger

Dr.-Ing. Martin Jung

Der 1972 gegründete Lehrstuhl Informatik 2 (Programmiersysteme) wird seit April 2002 von Prof. Dr. Michael Philippsen (als Nachfolger von Prof. Dr. H.-J. Schneider) geleitet. Eng mit dem Lehrstuhl assoziiert sind die beiden Professuren für Didaktik der Informatik und Open Source Software, deren Forschungsarbeiten separat dargestellt sind.

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 schon für Einprozessorsysteme und Arbeitsplatzrechner. Langfristiges Ziel ist es, den Anwendungen die verfügbare Rechen- und Kommunikationsleistung möglichst ungebremst zur Verfügung zu stellen bzw. aus begrenzten Systemen ein Maximum herauszuholen. Ein besonderer Arbeitsschwerpunkt sind Programmiersysteme für Multicore-Rechner, da deren unausweichliche Verbreitung ebenso wie die preisgünstige Verfügbarkeit von sehr leistungsstarker paralleler Spezialhardware im Massenmarkt (z.B. Grafik-Karten oder FPA-Hardware) kaum abschätzbare Auswirkungen auf die Software-Landschaft haben werden. Forschungsergebnisse werden stets an eigenen Prototypen und Demonstratoren praktisch evaluiert.

Wichtige Forschungsfelder

- **Vorhandenes Parallelisierungspotential ausschöpfen.** Da die Taktraten von Mehrkernrechnern kaum noch steigen, dafür aber deren Kernanzahl anwachsen wird, muss das Parallelisierungspotential existierender Software erkannt und ausgeschöpft werden, um an den Leistungssteigerungen der Hardware teilzuhaben. Außer vielleicht in Nischen führt kein Weg an einem Umstieg auf Parallelverarbeitung vorbei. Deshalb entwickelt der Lehrstuhl Werkzeuge, die dem Programmierer interaktiv beim Re-Engineering bestehender Anwendungen helfen, und erarbeitet Architekturmuster für neu zu entwickelnde Software-Projekte, die eine Skalierbarkeit für und damit eine Toleranz gegen wachsende Kernanzahlen aufweisen.
- **Hochleistungsanwendungen portabel machen.** Anwendungsprogrammierer erreichen nur dann bestmögliche Laufzeiten, wenn sie die Überwindung der Latenzzeiten und die explizite Kommunikation zwischen unterschiedlichen Komponenten der Systemarchitektur manuell angehen, ihren Code mit Hardware-nahen "Tricks" spezifisch für eine Architektur optimieren und ihre Applikation per Hand

in mehrere Teile zerlegen, von denen manche z.B. auf die Grafikkarte ausgelagert werden. Der Lehrstuhl erforscht, wie durch Anhebung des Abstraktionsniveaus der Programmierung die Produktivität gesteigert und die Portabilität verbessert werden kann, indem Code so übersetzt wird, dass verschiedene Teile auf heterogenen Systemkomponenten nebenläufig ausgeführt werden und dass die Datenkommunikation dazwischen transparent für den Entwickler erfolgt. Eine wichtige Frage dabei ist es, wie der Programmierer sein Wissen über bestehende Lokalisierungsbeziehungen programmiersprachlich ausdrücken kann, damit es effizienzsteigernd nutzbar bzw. damit Lokalität schon im Design sichtbar ist. Auf dem Weg zu diesem Ziel werden im Rahmen von Re-Engineering-Projekten Details der Hardware-Architektur vor dem Anwendungsentwickler verborgen, z.B. hinter Bibliotheksschnittstellen oder in domänenspezifischen Programmierspracherweiterungen.

- **Parallelitätsgrad dynamisieren.** Hochleistungsanwendungen werden oft für eine bestimmte Prozessorzahl entwickelt. Da die benötigten Rechnerbündelknoten vom Batch-System statisch für eine feste Zeitspanne zugeteilt werden, entstehen zwangsläufig unwirtschaftliche Reservierungslöcher. Analoge Probleme treten bei vielfädigen Anwendungen auf Multicore-Rechnern auf. Der Lehrstuhl arbeitet daher an der dynamischen Anpassung des Parallelitätsgrads durch Code-Transformationen, die die Kosten der resultierenden Datenumverteilungen berücksichtigen, sowie durch Interaktion und Zusammenarbeit mit dem Betriebssystem. Die in Programmen vorgefundenen expliziten, kontrollflussorientierten Synchronisationsmaßnahmen behindern die erforderlichen Analysen, weshalb der Lehrstuhl neue und bessere Programmkonstrukte erforscht, die die bisherigen adäquat ersetzen können und die Synchronisationserfordernisse deklarativ und an den Daten orientiert ausdrücken.
- **Parallelität testbar machen.** Im Software-Engineering nimmt Testen von jeher eine wichtige Stellung ein. Stichworte wie Testüberdeckung, Testdatengenerierung, Zuverlässigkeitsbewertung etc. gehören zum Handwerk. Leider berücksichtigt die Forschung in diesem Bereich den durch Nebenläufigkeit entstehenden Indeterminismus nur unzureichend. Der Lehrstuhl arbeitet daher an Werkzeugen zur Testdatengenerierung, die bei den zugrundeliegenden Überdeckungskriterien auch Verschränkungen nebenläufiger Programmäste berücksichtigen. Dies schließt Arbeiten an Betriebssystemschnittstellen und am Ablaufplaner ebenfalls ein. Weil ferner durch die Nebenläufigkeit der Suchraum in erheblichem Maße wächst, müssen Infrastrukturen erarbeitet werden, die es ermöglichen, Massentests auf großen Rechnerbündeln auszuführen.
- **Software-Entwicklungsprozesse verbessern.** Die heute in der Industrie praktizierte Entwicklung von komplexer, geschäfts- oder sicherheitskritischer Softwa-

re in global verteilten Teams verlangt nach der Einhaltung von wohldefinierten Software-Entwicklungsprozessen mit entsprechender Werkzeugunterstützung. Im Bereich der **praktischen Softwaretechnik** arbeitet der Lehrstuhl zusammen mit den **Honorar-Professoren Dr. Bernd Hindel und Dr. Detlef Kips**, die als Geschäftsführer zweier mittelständischer Software-Beratungsunternehmen über langjährige Praxiserfahrung in industriellen Software-Entwicklungsprojekten verfügen, daher an einer maschinen-ausführbaren Notation für die Modellierung von Software-Entwicklungsprozessen, wobei wegen der zum Einsatz kommenden vielfältigen Werkzeuge und Notationen, sowohl die (teil-) automatisierte Rückgewinnung von Nachverfolgbarkeitsinformation aus den Artefakten eines Entwicklungsprojekts, als auch die modellbasierte Entwicklung, Integration und Konfiguration von Softwarekomponenten betrachtet werden, wie sie insbesondere beim Entwurf eingebetteter Systeme für den Automobilbau üblich sind.

2. Forschungsprojekte

Die Belastung durch den gymnasialen Doppeljahrgang waren im Jahr 2011 so hoch, dass der Lehrstuhl auf eine detaillierte Darstellung der Einzelprojekte im Jahresbericht verzichten musste. Aktuelle Informationen können im Web gefunden werden unter <https://www2.informatik.uni-erlangen.de/research>

3. Publikationen

- Adersberger, Josef ; Philippsen, Michael: ReflexML: UML-based architecture-to-code traceability and consistency checking . In: Crnkovic, Ivica ; Gruhn, Volker ; Book, Matthias (Hrsg.) : Proceedings of the 5th European Conference on Software Architecture (ECSA 2011) (5th European Conference on Software Architecture (ECSA 2011) Essen 13.-16.09.2011). Berlin, Heidelberg : Springer-Verlag Berlin, Heidelberg, 2011, S. 344-359. - ISBN 978-3-642-23797-3
- Edelhäuser, Thorsten ; Philippsen, Michael ; Mutschler, Christopher: Trajectory Behavior Language . In: Artesis University College of Antwerp (Veranst.) : Proceedings of the 2nd International Conference on Positioning and Context-Awareness (PoCA 2011 Brussels 2011). 2011, S. 51-58. - ISBN 978-94-90705-04-6
- Ellner, Ralf ; Al-Hilank, Samir ; Drexler, Johannes ; Jung, Martin ; Kips, Detlef ; Philippsen, Michael: A FUML-Based Distributed Execution Machine for Enacting Software Process Models . In: France, Robert ; Kuester, Jochen ; Bordbar, Behzad ; Paige, Richard (Hrsg.) : Proceedings 7th European Conference on

- Modeling Foundations and Applications (Modelling Foundations and Applications Birmingham 06.-09.06.2011). Berlin / Heidelberg : Springer, 2011, S. 19-34. (Lecture Notes in Computer Science Bd. 6698) - ISBN 978-3-642-21469-1
- Kempf, Stefan ; Veldema, Ronald ; Philippsen, Michael: Is There Hope for Automatic Parallelization of Legacy Industry Automation Applications? In: GI (Hrsg.) : In Proceedings of the 24th Workshop on Parallel Systems and Algorithms (PARS 2011) (24th Workshop on Parallel Systems and Algorithms (PARS 2011) Rüşchlikon, Switzerland 26.-27.05.2011). 2011, S. 80-89.
 - Kempf, Stefan ; Veldema, Ronald ; Philippsen, Michael: Source Code Transformations to Increase the Performance of Software Transactional Memory (Poster Presentation) .Vortrag: 24th International Workshop on Languages and Compilers for Parallel Computing (LCPC 2011), Colorado, USA, 09.09.2011
 - Mutschler, Christopher ; Franke, Norbert ; Wolf, Daniel ; Witt, Nicolas: Apparatus and Method for Analyzing Sensor Data . Schutzrecht PCT/EP2011/069166 Patentanmeldung (31.10.2011)
 - Mutschler, Christopher ; Witt, Nicolas ; Philippsen, Michael ; Otto, Stephan: Apparatus and Method for Synchronizing Events . Schutzrecht PCT/EP2011/069160 Patentanmeldung (31.10.2011)
 - Mutschler, Christopher: Apparatus and Method for Transferring Event Detector Processes . Schutzrecht PCT/EP2011/069159 Patentanmeldung (31.10.2011)
 - Mutschler, Christopher: Apparatus and Method for Transmitting A Message to Multiple Receivers . Schutzrecht PCT/EP2011/069169 Patentanmeldung (31.10.2011)
 - Mutschler, Christopher ; Kókai, Gabriella ; Edelhäuser, Thorsten: Online Data Stream Mining on Interactive Trajectories in Soccer Games . In: Artesis University College of Antwerp (Veranst.) : Proceedings of the 2nd International Conference on Positioning and Context-Awareness (PoCA 2011 Brussels 2011). 2011, S. 15-22. - ISBN 978-94-90705-04-6
 - Mutschler, Christopher ; Otto, Stephan ; Witt, Nicolas: Portable Device, Virtual Reality System and Method . Schutzrecht EP11183970.0 Patentanmeldung (05.10.2011)
 - Oster, Norbert ; Philippsen, Michael: Structural Equivalence Partition and Boundary Testing . In: Reussner, Ralf ; Grund, Matthias ; Oberweis, Andreas ; Tichy, Walter (Hrsg.) : Proceedings 183 Software Engineering 2011 - Fachtagung des GI-Fachbereichs Softwaretechnik (Software Engineering 2011 - Fachtagung des

GI-Fachbereichs Softwaretechnik Karlsruhe 21.02.-25.02.2011). Bd. 183. Karlsruhe : Köllen, 2011, S. 75-86. (Lecture Notes in Informatics Bd. 183) - ISBN 978-3-88579-277-2

- Philippsen, Michael ; Tillmann, Nikolai ; Brinkers, Daniel: Double inspection for run-time loop parallelization . In: ' (Hrsg.) : Proceedings of the 24th International Workshop on Languages and Compilers for Parallel Computing (LCPC 2011) (International Workshop on Languages and Compilers for Parallel Computing (LCPC 2011) Colorado, USA 08.-10.09.2011). 2011, S. '.
- Tausch, Norbert ; Philippsen, Michael ; Adersberger, Josef: A Statically Typed Query Language for Property Graphs . In: Bernardino, Jorge ; Cruz, Isabel ; Desai, Bipin C. (Hrsg.) : Proceedings of 15th International Database Engineering and Applications Symposium (IDEAS'11) (15th International Database Engineering and Applications Symposium (IDEAS'11) Lissabon, Portugal 21.-23.09.2011). online : ACM, 2011, S. 219-225. - ISBN 978-1-4503-0627-0
- Veldema, Ronald ; Philippsen, Michael: A Hybrid Functional and Object-Oriented Language for a Multi-Core Future . In: it-Information Technology (ISSN 1611-2776) 53 (2011), Nr. 2, S. 84-90
- Veldema, Ronald ; Blaß, Thorsten ; Philippsen, Michael: Enabling Multiple Accelerator Acceleration for Java/OpenMP . In: McCool, Michael ; Rosenblum, Mendel (Hrsg.) : Proceedings 3rd USENIX Workshop on Hot Topics in Parallelism (HotPar '11) (3rd USENIX Workshop on Hot Topics in Parallelism (HotPar '11) Berkeley, CA 26.-27.05.2011). 2011, S. 1-6.
- Veldema, Ronald ; Philippsen, Michael: Iterative data-parallel mark & sweep on a GPU . In: Boehm, Hans-Juergen ; Bacon, David F. (Hrsg.) : Proceedings of the International Symposium on Memory Management (International Symposium on Memory Management (ISMM '11) San Jose, California, USA). New York : ACM, 2011, S. 1-10. - ISBN 978-1-4503-0263-0
- Weigend, Johannes ; Siedersleben, Johannes ; Adersberger, Josef: Dynamische Analyse mit dem Software-EKG . In: Reussner, Ralf ; Grund, Matthias ; Oberweis, Andreas ; Walter, Tichy (Hrsg.) : Proceedings 183 Software Engineering 2011 - Fachtagung des GI-Fachbereichs Softwaretechnik (Software Engineering 2011 - Fachtagung des GI-Fachbereichs Softwaretechnik Karlsruhe 21.02.-25.02.2011). Bd. 183. Karlsruhe : Köllen, 2011, S. 3-18. (Lecture Notes in Informatics Bd. 183) - ISBN 978-3-88579-277-2
- Werth, Tobias ; Schreier, Silvia ; Philippsen, Michael: CellCilk: Extending Cilk for heterogeneous multicore platforms . In: ' (Hrsg.) : Proceedings of the 24th

International Workshop on Languages and Compilers for Parallel Computing (LCPC 2011) (24th International Workshop on Languages and Compilers for Parallel Computing (LCPC 2011) Colorado, USA 08.-10.09.2011). 2011, S. '.

4. Studien- und Abschlussarbeiten

- Diplomarbeit: Erweiterung des modellbasierten Testdatengenerators AFRA CASE um die Unterstützung von Skriptsprachen und Verbesserung der Performance bei der Behandlung von Verzweigungen. Bearbeiter: Chuanru Ma (beendet am 14.01.2011); Betreuer: Prof. Dr. Michael Philippsen; Dr.-Ing. Norbert Oster
- Diplomarbeit: Entwicklung und Realisierung eines Konzepts zur Qualitätssicherung von (Code-) Generatoren im MDSD-Kontext. Bearbeiter: Thomas Fischer (beendet am 16.01.2011); Betreuer: Dipl.-Inf. Ralf Ellner; Dr.-Ing. Norbert Oster; Prof. Dr. Michael Philippsen
- Bachelor Thesis: Plattformunabhängige Parallelisierung des Lattice-Boltzmann-Verfahrens mit OpenCL. Bearbeiter: Carolin Wolf (beendet am 17.01.2011); Betreuer: Dipl.-Inf. Tobias Werth; Prof. Dr. Michael Philippsen
- Diplomarbeit: Portierung eines Ansatzes zur Unterstützung der automatischen Datenfluss- und Bedingungsüberdeckung beim Testen von C / C++ / C# - Programmen. Bearbeiter: Eugen Wagner (beendet am 15.03.2011); Betreuer: Dr.-Ing. Norbert Oster; Prof. Dr. Michael Philippsen
- Diplomarbeit: Compilerunterstützte Parallelisierung des Lattice-Boltzmann-Verfahrens mit OpenCL. Bearbeiter: Dimitrij Kotrev (beendet am 28.04.2011); Betreuer: Dipl.-Inf. Tobias Werth; Prof. Dr. Michael Philippsen
- Studien-/Bachelor-/Diplom-/Masterarbeit: A Tool-Chain to Generate a Context Sensitive Process Documentation. Bearbeiter: Halimatou Poussami (beendet am 13.05.2011); Betreuer: Dipl.-Inf. Ralf Ellner; Hon.-Prof. Dr.-Ing. Detlef Kips
- Bachelor Thesis: Entwurf und Implementierung eines Plugin-Frameworks zur Quellcode Analyse. Bearbeiter: Drescher Matthias (beendet am 02.08.2011); Betreuer: Dipl.-Inf. Georg Dotzler; Prof. Dr. Michael Philippsen
- Bachelor Thesis: Erweiterung eines C-Compilers zur Codeerzeugung für Systeme der Industrieautomatisierung. Bearbeiter: Christoph Romstöck (beendet am 23.10.2011); Betreuer: Dipl.-Inf. Stefan Kempf; Prof. Dr. Michael Philippsen

- Diplomarbeit: Erhebung und Analyse von Kennzahlen aus dem fachlichen Performance-Monitoring. Bearbeiter: Stefan Eberlein (beendet am 14.11.2011); Betreuer: M. Eng. Norbert Tausch, Dipl.-Ing. (FH); Dr.-Ing. Norbert Oster; Prof. Dr. Michael Philippsen