UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft

Annual Report 2010/2011

PADERBORN
CENTER FOR
PARALLEL
COMPUTING

University of Paderborn
Paderborn Center for Parallel Computing
Fürstenallee 11, D-33102 Paderborn

www.uni-paderborn.de/pc2

Table of Contents

1 Foreword

The Paderborn Center for Parallel Computing (PC^2) is the University of Paderborn's central focus for connecting research and services in High Performance Computing (HPC). Our provide access to our high-performance systems operate by us; we also operate corresponding computers on behalf of individual working groups of the University of Paderborn. At the same time, we strive to enrich these services by our research results – research in resource management in data centers, hardware and software integration, or energy-efficient operations.

The PC^2 draws on the support of the advisory and the executive board in its work. With the appointment of Prof. Walther and Prof. Vrabec, further expertise from mathematics and mechanical engineering could be added to the board. While we were glad to gain new board members, at the same time had to let other long-time members go.

Dr. Sascha Effert and Dr. Tobias Schumacher successfully completed their PhD in 2011, and Mr. Nils Lücking finished his training as an IT specialist in June 2011.

Among all these positive reports, unfortunately two tragic incidents need to be mentioned. A fatal accident and a serious illness within PC^2's members deeply affected everybody and left a void.

This annual report depicts our work and the most important results from 2010 and 2011. These two years have been shaped especially by the highly successful work of junior professor André Brinkmann, executive head of PC^2. Mr. Brinkmann has been highly successful in project acquisition from industrial as well as public project sponsors, starting shortly after his appointment in July of 2008 and continuously thereafter. Therefore, it came as no surprise when, in October of 2011, Dr. Brinkmann was appointed a W3 chair at the University of Mainz and entrusted with the management of their Data Processing Center. We congratulate Dr. Brinkmann to this important career move and wish him all the best for the future. Next to Mr. Brinkmann, Prof. Marco Platzner and junior professor Christian Plessl have been especially successful in project fundraising.

The EU-projects EPiCS (Engineering Proprioception in Computing Systems) and EDGI (European Desktop Grid Initiative) as well as the ENHANCE project (Enabling Heterogeneous Hardware Acceleration Using Novel Pogramming and Scheduling Models), which is funded by the BMBF, fall into this reporting interval.

Additionally, the PC^2 has successfully acquisitioned the BMWi projects Simba (Simulation Backbone Automotive) and GreenPAD – energy-optimized ICT for regional economic and science cluster.

Next to these publicly sponsored projects described in this annual report, the year 2011 has been characterized by another important milestone: a grant application for a new mainframe computer according to the funding regulations of constitution §91b. The University's support – constructing building O including a modern infrastructure for a datacenter – and the support of the Land Northrhine-Westfalia made it possible for us to file an application for a total of 4 Million Euros. And – even though it exceeds the timeframe of this report – the vital news: early in 2012, the grant was fully approved without any cuts. The next steps will now be the configuration, tendering, acquisition, and commissioning of this new high-performance computer. The new system is expected to be about 20 times faster than the old one. We are looking forward to this new task and to offering a modern, innovative environment for our users and researchers, probably by the end of 2012.

Such a powerful computer system could have never been operated in the old PC² computer room at Fürstenallee. Therefore, moving from the Heinz Nixdorf building to the newly constructed building O on the University campus, in the end of 2011, was imperative. The new surroundings offer a wide range of possibilities for further growth not only for HPC systems but also for all PC^2 members, old and new. Altogether we can once again look back on two successful years. We are confident to be able to continue this success in the coming years.

Prof. Dr. Holger Karl
Chairman of the PC² board
August 2012

Vorwort

Die Universität Paderborn unterhält mit dem Paderborn Center for Parallel Computing (PC^2) eine zentrale wissenschaftliche Einrichtung, die Forschung und Dienstleistungen im Hochleistungsrechnen (High Performance Computing, HPC) miteinander verbindet. Als Dienstleistung stellen wir den Zugang zu eigenen Systemen zur Verfügung; wir betreiben auch entsprechende Rechner im Auftrag einzelner Arbeitsgruppen der Universität Paderborn. Dabei sind wir bestrebt, unsere Dienstleistungen durch unsere Forschungsergebnisse zu bereichern – Forschung auf den Gebieten der Ressourcenverwaltung in Rechenzentren, der Hardware-/Software-Integration oder des energieeffizienten Betriebes.
Das PC^2 kann in seiner Arbeit auf die Unterstützung durch den Beirat und den Vorstand zurückgreifen. Mit der Bestellung von Frau Prof. Walther und Herrn Prof. Vrabec erhielt der Vorstand weiteren Sachverstand aus der Mathematik und dem Maschinenbau. Wir sind sehr froh, dass wir wieder neue Mitglieder für unseren Vorstand dazu gewinnen konnten, leider mussten wir uns aber auch von einigen langjährigen Mitgliedern verabschieden.

Herr Dr. Sascha Effert und Herr Dr. Tobias Schumacher haben 2011 erfolgreich ihre Promotion abgelegt und Herr Nils Lücking hat seine Ausbildung zum Fachinformatiker im Juni 2011 erfolgreich abgeschlossen.

Neben all den positiven Berichtsinhalten sind auch leider zwei tragische Ereignisse zu erwähnen. Ein Unfalltod und eine schwere Erkrankung im Kreise der PC² Mitarbeiter haben jeden tief getroffen und eine Lücke hinterlassen.

Dieser Jahresbericht zeichnet unsere Tätigkeiten und die wichtigsten Ergebnisse aus den Jahren 2010 und 2011 nach. Diese beiden Jahre waren insbesondere durch die sehr erfolgreiche Tätigkeit des geschäftsführenden Leiters des PC^2, Herrn Juniorprofessor André Brinkmann geprägt. Herr Brinkmann war bereits kurz nach seiner Berufung im Juli 2008 und auch stetig danach sehr erfolgreich in der Projektakquisition, sowohl bei industriellen wie auch öffentlichen Projektförderern. Dies führte dann im Oktober 2011 zu dem nahezu zwangsläufigen Resultat: Herr Brinkmann wurde auf eine W3-Professur an die Universität Mainz berufen und dort mit der Leitung des Zentrums für Datenverarbeitung betraut. Wir gratulieren Herrn Brinkmann zu diesem wichtigen Karriereschritt und wünschen ihm für die Zukunft alles Gute. Neben Herrn Brinkmann waren insbesondere auch Prof. Marco Platzner und Juniorprofessor Christian Plessl sehr erfolgreich bei Projekteinwerbungen.
In der Berichtsperiode sind die EU-Projekte EPiCS (Engineering Proprioception in Computing Systems) und EDGI (European Desktop Grid Initiative) sowie das von dem

BMBF geförderte ENHANCE Projektes (Enabling Heterogeneous Hardware Acceleration Using Novel Pogramming and Scheduling Models) zu nennen.
Zudem war das PC^2 erfolgreich bei der Einwerbung von Projekten des BMWi mit den Projekten Simba (Simulation Backbone Automotive) und GreenPAD – Energieoptimierte IKT für regionale Wirtschafts- und Wissenscluster.
Neben diesen geförderten Projekten, die in diesem Jahresbericht umfassend beschrieben sind, war das Jahr 2011 auch durch einen wichtigen Meilenstein geprägt: Die Einreichung eines Förderantrages für einen neuen Großrechner nach den Förderbestimmungen des Grundgesetzes §91b. Durch die Unterstützung der Hochschule – mit dem Bau des Gebäudes O samt einer modern ausgestatteten Infrastruktur für ein Rechenzentrum – und des Landes NRW war es uns möglich, einen Antrag mit einem Gesamtvolumen von 4 Millionen Euro zu stellen. Und – auch wenn dies nicht in den Zeitraum dieses Bericht fällt – so doch die entscheidende Nachricht: Anfang 2012 wurde dieser Antrag vollständig, ohne Kürzungen genehmigt. Die nächsten Schritte werden nun die Konfiguration, Ausschreibung, Beschaffung und Inbetriebnahme dieses neuen Hochleistungsrechners sein. Die zu erwartende Systemleistung ist ungefähr um Faktor 20 größer als die der bisherigen Rechnersysteme. Wir freuen uns sehr auf diese Aufgabe und darauf, unseren Nutzern und Forschern eine moderne, innovative Umgebung voraussichtlich zum Ende des Jahres 2012 bereitstellen zu können.
Ein derart leistungsfähiges Rechnersystem hätte niemals im alten PC²-Rechnerraum in der Fürstenallee betrieben werden können. Deshalb wurde der Umzug aus dem Heinz Nixdorf Gebäude in das neuerstellte Gebäude O auf dem Universitätscampus zum Ende des Jahres 2011 zwingend notwendig. Nicht nur die HPC-Systeme, sondern auch die Mitarbeiter des PC² finden in den neuen Räumlichkeiten eine sehr gute Umgebung vor, die genügend Möglichkeiten zur weiteren Entwicklung offen hält.

Insgesamt können wir wieder auf zwei erfolgreiche Jahre zurückblicken. Wir sind zuversichtlich, dass dies auch in den nächsten Jahren fortgesetzt werden kann.

Prof. Dr. Holger Karl
Vorsitzender des Vorstandes PC²
Im August 2012

2 Inside PC²

2.1 Board

The PC² is headed by an interdisciplinary board comprising professors from various working groups. The following people were assigned to the PC^2 board in the reporting period.

2.2 Members of the Board

Prof. Dr. Holger Karl (Chairman)
Faculty of Electrical Engineering, Computer Science and Mathematics

Jun.-Prof. Dr.-Ing. André Brinkmann (until October 2011)
Faculty of Electrical Engineering, Computer Science and Mathematics

Prof. Dr. Michael Dellnitz
Faculty of Electrical Engineering, Computer Science and Mathematics

Prof. Dr. Gregor Fels
Faculty of Science

Prof. Dr. Burkhard Monien
Faculty of Electrical Engineering, Computer Science and Mathematics

Dr. Gudrun Oevel
Zentrum für Informations- und Medientechnologien (IMT) representative

Prof. Dr. Marco Platzner
Faculty of Electrical Engineering, Computer Science and Mathematics

Prof. Dr. Franz Josef Rammig
Faculty of Electrical Engineering, Computer Science and Mathematics

Prof. Dr. Wolf Gero Schmidt
Faculty of Theoretical Physics

Prof. Jadran Vrabec (since 2011)
Thermodynamics and Energy Technology

Prof. Andrea Walther (since 2011)
Faculty of Electrical Engineering, Computer Science and Mathematics

Prof. Dr. Hans-Joachim Warnecke
Faculty of Science

Dr. Jens Simon
Paderborn Center for Parallel Computing
Assistant researchers´ representative

Dipl.-Inform. Axel Keller
Paderborn Center for Parallel Computing
Non researchers´ representative

Lars Schäfers (since 2011)
Assistant researchers´representative

Jörn Tilmanns (since 2011)
Student representative

2.3 PC² Advisory Board

Karsten Beins
Senior Director Portfolio & Technology
Fujitsu Technology Solutions, Paderborn

Dr. Horst Joepen
Chief Executive Officer
Searchmetrics GmbH, Berlin

Prof. Dr. Dr. Thomas Lippert
Director of Institute for Advanced Simulation, Head of Jülich Supercomputing Centre,
Jülich Supercomputing Centre, Jülich

Prof. Dr. Alexander Reinefeld
Head of Computer Science, Zuse-Institut Berlin
Humboldt Universität Berlin

Prof. Dr.-Ing. Michael Resch
Director of the High Performance Computing Center Stuttgart
HLRS Höchleistungsrechenzentrum Stuttgart

Prof. Dr. Nikolaus Risch
President of the University of Paderborn
Universität Paderborn

Dr. Werner Sack,
Miele, Gütersloh

2.4 PC² Staff

The following people were assigned to the PC² for the period of time covered by this report.

Jannic Altstädt (Trainee since August 2010)
Dipl.-Inform. Bernard Bauer
M.Sc. Tobias Beisel
Fabian Berendes (Trainee since August 2011)
Dipl.-Inform. Georg Birkenheuer
Jun.-Prof. Dr.-Ing. André Brinkmann (until September 2011)
Dipl.-Inform. Sascha Effert (until June 2011)
Birgit Farr (Secretary)
M.Sc. Ramy Gad (since October 2011)
Dipl.-Inform. Yan Gao
M.Sc. Mariusz Grad (until October 2011)
M.Sc. Matthias Grawinkel
Dipl.-Inform. Jürgen Kaiser (since May 2011)
Dipl.-Inform. Dipl.-Math. Paul Kaufmann
Dipl.-Inform. Axel Keller
Dipl.-Inform. Matthias Keller (until December 2011)
Michaela Kemper (Secretary)
Dipl.-Inform. Tobias Kenter
Dipl.-Ing. Andreas Krawinkel
Dipl.-Inform. Jens Lischka (until June 2010)
Dipl.-Ing. Enno Lübbers (until June 2010)
Nils Lücking (until October 2011)
Dipl.-Inform Fabio Margaglia (since September 2010)
M.Sc. Dirk Meister
Dipl.-Ing. Björn Meyer
Dr. Lars Nagel (since October 2011)
Dipl.-Inform. Oliver Niehörster
Holger Nitsche
Dr. Christian Plessl (until September 2011)
Dipl.-Inform Ivan Popov (since September 2010)
Dipl.-Inform. Lars Schäfers
Dipl.-Inform. Tobias Schumacher (until June 2011)
Dr. Jens Simon

Within the reporting period additional support was provided by students and graduate assistants who were engaged part time (9.5 h/week and 19 h/week) in tasks, which included programming, user support, system administration, etc.

Zahra Aghayary	Matthias Bolte	Tobias Bertel
Christoph Bröter	Hubert Dömer	Martin Dräxler
Denis Dridger	Matthias Frye	Viktor Gottfried
Gokul K. Gunasekaran	A. Abhishek Hanagodu	Christoph Kleineweber
Alexander Krieger	Marcel Lauhoff	Frank Ingo Meith
Markus Pargmann	Inga Poste	Christoph Raupach
Kai-Uwe Renken	Jörn Schumacher	Johannes Schuster
Elmar Weber	Tobias Wiersema	Adrian Wilke

In the year 2010/2011 the PC² employed two trainees to learn the trade of a “computer specialist” (Fachinformatiker) in the field of system integration. With the source required to employ trainees provided by the North Rhine-Westphalia government, the PC² was able to oversee this priority assignment.

3 Research and Projects

3.1 Research Areas

Research interests of the PC² are parallel and distributed large scale systems.
The focus of research is on

- Custom Computing & Many Cores
- Middleware & system software
- Scalable Storage Systems
- Testbeds & Benchmarking

Current information is also presented on the web pages of the PC²
(http:/www.upb.de/pc2).

Computer Architecture
Application-specific coprocessors can significantly accelerate many high-performance computing (HPC) applications. Designing fast accelerators and optimizing their performance remains a difficult task requiring significant hardware design expertise.
The PC² has a long-term experience with innovative cluster systems based on commodity as well as on specialized hardware components. Different techniques to accelerate compute nodes are considered, like multi-core processors, graphical processing units (GPUs), and acceleration cards equipped with field-programmable gate arrays (FPGAs). Hence, vendors of supercomputers and high-performance workstations are beginning to integrate reconfigurable accelerators in their products, which makes this custom computing technology available to a broader user group. One of our missions is to make the potential of custom computing more accessible to users. To this end, we work on basic infrastructure for reconfigurable computers, specifically we work on flexible and portable communication infrastructures and on runtime systems that support dynamic reconfiguration. In several application projects we are exploring the applicability of these infrastructures by building scalable accelerators for HPC applications that exploit the performance of FPGAs and standard CPUs.

Maximizing the performance of an application consisting of many tasks is challenging, since the hardware accelerator cores affect each other when accessing shared resources. Hence, meticulous care has to be taken to avoid bottlenecks in an implementation. To support the designer with performance optimization, we are working on estimating the

application's performance with a model-based approach. By combining a model of the application and a model of the execution architecture, we can study the influence of various system parameters, such as communication bandwidths and latencies, and can use this information for performance optimization

Research Topics	***Contact***	***Email***
Heterogeneous computing systems	Tobias Beisel	tbeisel@upb.de
CPU-accelerator architectures	Tobias Kenter	kenter@upb.de
High performance custom computing	Jun.-Prof. Dr. Christian Plessl	Christian.plessl@ub.de
Computer system architecture	Dr. Jens Simon	simon@upb.de

Middleware & system software

The PC²'s research focuses on the problem of how to guarantee Service Level Agreements (SLA) in Cloud and Grid environments. This research includes fault tolerance mechanisms like checkpointing and migration of jobs and the assessment of the likelihood of SLA violations. The combined instruments, risk assessment and fault tolerance mechanisms, allow a powerful risk aware management of cluster, cloud, and grid jobs. This improves the guaranteed service quality of the resource management.

In addition, the PC² works on the integration of Web service based Enterprise Application Integration (EAI) into Grid and Cloud environments. Our aim is to combine the strengths of the two areas, loosely coupled services and secure and easy to deploy grid infrastructures. The result will be the ability to create secure business workflows in distributed infrastructures.

Resource Management Systems (RMS) are needed for the grid as well as for compute clusters. They allow users and system administrators to access and manage various computing resources like processors, memory, networks, or storage. PC² has developed an expandable and modular RMS, called Computing Center Software (openCCS), which uses a planning based job scheduler. OpenCCS is used in several projects and its features are continuously extended.

Research Topics	*Contact*	*Email*
Risk Assessment / Management, Service Level Agreements,	Georg Birkenheuer	birke@upb.de
Grid and Cloud Computing	Matthias Keller	mkeller@upb.de
Grid-based integration and orchestration of business information systems	Holger Nitsche	hn@upb.de
Computing Center Software (OpenCCS)	Axel Keller	kel@upb.de

Scalable Storage Systems

Data has become one of the most valuable assets in all businesses and one of the most volatile. With all businesses, today, the growth of data needed to operate is increasing at 67% per year at current business practices. To worsen the data storage problem, many new regulations require that much data and electronic records and emails be readily available. For these reasons, the requirement for storage continues to grow at a phenomenal pace. Complexity and the proprietary nature of storage systems have meant and continue to mean high investment and management costs.

The Paderborn Center for Parallel Computing develops parallel storage algorithms, integrates them in scalable storage architectures and leverages their usage by new management concepts.

Research Topics	*Contact*	*Email*
Scalable Storage Systems	Sascha Effert	fermat@douglas2a.de
Parallel Data Deduplication	Dirk Meister	dirkmeister@uni-mainz.de
Archiving	Matthias Grawinkel	grawinkel@uni-paderborn.de

Testbeds & Benchmarking

The development of software components for highly complex networked systems requires, besides analytical and simulation-based evaluation methods, more and more experiments in large real live traffic environments. One method to build a new system on top of an existing system is to use virtualization. Virtualization of resources can be found in all areas of computing. Also in the domain of networking, virtualization is used to hide the characteristics of network resources (like routers, switches, etc.) from the way in which other systems interact with them. The PC² is engaged in investigating how virtualization can be utilized as a concept in the context of building new network testbeds.

The PC² benchmarking center is specialized in evaluating the performance of high-speed networks and parallel computer systems. Typically, these are based on cluster technology. We evaluate functional parts or complete systems with the help of so-called low-, system-, and application-level benchmarks. Derived from this evaluation new system architectures will be developed. In addition, the PC² offers assistance with running existing parallel applications in a cost efficient way and with porting applications to high performance parallel computers.

Research Topics	***Contact***	***Email***
System Evaluation, Benchmarking, Experimental Cluster Systems	Dr. Jens Simon	simon@uni-paderborn.de

3.2 Projects

Projects started in 2010 and 2011

Funding agency	*Projectname*	*Start of the Project*	*End of the Project*
EU	EDGI	June 2010	May 2012
EU	EPiCS	September 2010	August 2014
BMBF	Enhance	April 2011	September 2013
BMWi	GreenPAD	June 2011	May 2014
BMWi - ZIM	Simba	April 2011	March 2013

Ongoing Projects

Funding agency	*Projectname*	*Start of the Project*	*End of the Project*
EU	Scalus	December 2009	November 2013
BMBF	DGSI	May 2009	April 2012
BMBF	MosGrid	September 2009	August 2012
Microsoft Research Ltd.	GOmputer	March 2009	February 2012

Projects finished in 2010 and 2011

Funding agency	*Projectname*	*Start of the Project*	*End of the Project*
EU	Hydra	October 2008	June 2010
EU	OneLab2	September 2008	November 2010
BMWi	RECS	January 2009	July 2010
BMWi	Prothesengeschäft	April 2009	March 2011
BMWi	ProAdapt-2	July 2009	October 2011
BMWi	Tumordiagnose	July 2009	June 2011

For current information about our projects please refer to our website.

3.3 Publications, Grants, and Awards

Papers 2010

Mariusz Grad and Christian Plessl
Pruning the Design Space for Just-in-time Processor Customization
Proceedings of the International Conference on Reconfigurable Computing, 2010.

Mariusz Grad and Christian Plessl
Pivpav: An Open source Circuit Library with Benchmarking Facilities
Proceedings of the International Conference on Engineering of Reconfigurable Systems and Algorithms, pp. 144-150, 2010.

Paul Lensing, Dirk Meister and André Brinkmann
hashFS: Applying Hashing to Optimized File Systems for Small File Reads
Proceedings of the 6th International Workshop on Storage Network Architecture and Parallel I/Os (SNAPI'10), 2010.

Petra Berenbrink, Andre Brinkmann, Tom Friedetzky and Lars Nagel
Balls into Non-uniform Bins
Proceedings of the 24th IEEE International Parallel & Distributed Processing Symposium (IPDPS), 2010.

Georg Birkenheuer, Sebastian Breuers, André Brinkmann, Dirk Blunk, Gregor Fels, Sandra Gesing, Sonja Herres-Pawlis, Oliver Kohlbacher, Jens Krüger and Lars Packschies
Grid-Workflows in Molecoular Science
Proceedings of the Grid Workflow Workshop (GWW), 2010.

Dirk Meister and Andre Brinkmann
dedupv1: Improving Deduplication Throughput Solid State Drives (SSD)
Proceedings of the 26th IEEE Symposium on Massive Storage Systems and Technologies (MSST), 2010.

Georg Birkenheuer, André Brinkmann and Holger Karl
Risk Aware Overbooking for Commercial Grids
Proceedings of the 15th Workshops on Job Scheduling Strategies for Parallel Processing (JSSPP), 2010.

Matthias Bolte, Michael Sievers, Georg Birkenheuer, Oliver Niehörster and André Brinkmann
Non-intrusive Virtualization Management Using libvirt
Design, Automation, and Test in Europe (DATE), 2010.

André Brinkmann, Dominic Battré, Georg Birkenheuer, Odej Kao and Kerstin Voß
Risikomanagement für verteilte Umgebungen
Forschungs Forum Paderborn (FFP), no. 13, 2010.

Yan Gao, Dirk Meister and André Brinkmann
Reliability Analysis of Declustered-Parity RAID 6 with Disk Scrubbing and Considering Irrecoverable Read Errors
The 5th IEEE International Conference on Networking, Architecture, and Storage (NAS 2010), 2010.

Tobias Kenter, Marco Platzner, Christian Plessl and Michael Kauschke
Performance Estimation for the Exploration of CPU-Accelerator Architectures
accepted for publication in: Workshop on Proc. Workshop on Architectural Research Prototyping (WARP), 2010.

Petra Berenbrink, André Brinkmann, Tom Friedetzky and Lars Nagel
Balls into Bins with Related Random Choices
Proceedings of the 22nd ACM Symposium on Parallelism in Algorithms and Architectures (SPAA), 2010.

David Andrews and Christian Plessl
Configurable Processor Architectures: History and Trends
to appear in: Proc. Int. Conf. on Engineering of Reconfigurable Systems and Algorithms (ERSA), CSREA Press, 2010.

Enno Lübbers, Marco Platzner, Christian Plessl, Ariane Keller and Bernhard Plattner
Towards Adaptive Networking for Embedded Devices based on Reconfigurable Hardware
to appear in: Proc. Int. Conf. on Engineering of Reconfigurable Systems and Algorithms (ERSA), CSREA Press, 2010.

Mariusz Grand and Christian Plessl
An Open Source Circuit Library with Benchmarking Facilities
to appear in: Proc. Int. Conf. on Engineering of Reconfigurable Systems and Algorithms (ERSA), CSREA Press, 2010.

Matthias Woehrle, Christian Plessl and Lothar Thiele
Rupeas: Ruby Powered Event Analysis DSL
to appear in: Proc. Int. Conf. Network Sensing Systems (INSS), 2010.

Tobias Beisel, Manuel Niekamp and Christian Plessl
Using Shared Library Interposing for Transparent Acceleration in Systems with Heterogeneous Hardware Accelerators
to appear in: Proc. IEEE Int. Conf. on Application-Specific Systems, Architectures, and Processors (ASAP), 2010.

Neeli R. Prasad, Markus Eisenhauer, Matts Ahlsén, Atta Badii, André Brinkmann, Klaus Marius Hansen and Peter Rosengren
Open Source Middleware for Networked Embedded Systems towards Future Internet of Things
Vision and Challenges for Realising the Internet of Things, 4.8, pp. 153 - 163, 2010.

Yan Gao, Dirk Meister and André Brinkmann
Request Balancing on SkewCCC
2nd International Workshop on DYnamic Networks: Algorithms and Security (Dynas), 2010.

Oliver Niehörster, André Brinkmann, Gregor Fels, Jens Krüger and Jens Simon
Enforcing SLAs in Scientific Clouds
IEEE International Conference on Cluster Computing 2010 (Cluster), 2010.

Sandra Gesing, Istvan Marton, Georg Birkenheuer, Bernd Schuller, Richard Grunzke, Jens Krüger, Sebastian Breuers, Dirk Blunk, Gregor Fels, Lars Packschies, Andre Brinkmann, Oliver Kohlbacher and Miklos Kozlovszky
Workflow Interoperability in a Grid Portal for Molecular Simulations
Proceedings of the International Workshop on Scientific Gateways 2010 (IWSG), pp. 44-48, 2010.

Martin Wewior, Lars Packschies, Dirk Blunk, Daniel Wickeroth, Klaus-Dieter Warzecha, Sonja Herres-Pawlis, Sandra Gesing, Sebastian Breuers, Jens Krüger, Georg Birkenheuer and Ulrich Lang
The MoSGrid Gaussian Portlet - Technologies for the Implementation of Portlets for Molecular Simulations
International Workshop on Scientific Gateways 2010 (IWSG'10), 2010.

Marcin Bienkowski, Andre Brinkmann, Marek Klonowski and Miroslaw Korzeniowski
SkewCCC+: A Heterogeneous Distributed Hash Table
Proceedings of the 14th International Conference On Principles Of Distributed Systems (Opodis), 2010.

Papers 2011

Christoph Kleineweber, Axel Keller, Oliver Niehörster and Andre Brinkmann
Rule Based Mapping of Virtual Machines in Clouds
Proceedings of the 19th Euromicro International Conference on Parallel, Distributed and Network-Based Computing (PDP), 2011.

Georg Birkenheuer, Dirk Blunk, Sebastian Breuers, André Brinkmann, Gregor Fels, Sandra Gesing, Richard Grunzke, Sonja Herres-Pawlis, Oliver Kohlbacher, Jens Krüger, Ulrich Lang, Lars Packschies, Ralph Müller-Pfefferkorn, Patrick Schäfer, Johannes Schuster, Thomas Steinke, Klaus-Dieter Warzecha and and Martin Wewior
MoSGrid: Progress of Workflow driven Chemical Simulations
Proceedings of the Grid Workflow Workshop (GWW), 2011.

Sandra Gesing, Peter Kacsuk, Miklos Kozlovszky, Georg Birkenheuer, Dirk Blunk, Sebastian Breuers, Andre Brinkmann, Gregor Fels, Richard Grunzke, Sonja Herres-Pawlis, Jens Krüger, Lars Packschies, Ralf Müller-Pfefferkorn, Patrick Schäfer, Thomas Steinke, Anna Szikszay Fabri, Klaus Warzecha, Martin Wewior and Oliver Kohlbacher
A Science Gateway for Molecular Simulations
In: EGI User Forum 2011, Book of Abstracts, pp. 94–95, 2011.

Sandra Gesing, Richard Grunzke, Ákos Balaskó, Georg Birkenheuer, Dirk Blunk, Sebastian Breuers, André Brinkmann, Gregor Fels, Sonja Herres-Pawlis, Peter Kacsuk, Miklos Kozlovszky, Jens Krüger, Lars Packschies, Patrick Schäfers, Bernd Schuller, Johannes Schuster, Thomas Steinke, Anna Szikszay Fabri, Martin Wewior, Ralph Müller-Pfefferkorn and Oliver Kohlbacher
Granular Security for a Science Gateway in Structural Bioinformatics
Proceedings of the International Workshop on Scientific Gateways 2010 (IWSG), 2011.

Matthias Grawinkel, Thorsten Schäfer, Andre Brinkmann, Jens Hagemeyer and Mario Porrmann
Evaluation of Applied Intra-Disk Redundancy Schemes to Improve Single Disk Reliability
Proceedings of the 19th Annual Meeting of the IEEE International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (Mascots), 2011.

Andre Brinkmann, Yan Gao, Miroslaw Korzeniowski and Dirk Meister
Request Load Balancing for Highly Skewed Traffic in P2P Networks
Proceedings of 6th IEEE International Conference on Networking, Architecture, and Storage (NAS), 2011.

Oliver Niehörster, Axel Keller and André Brinkmann
An Energy-Aware SaaS Stack
Proceedings of the 19th Annual Meeting of the IEEE International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS), 2011.

Oliver Niehörster, Alexander Krieger, Jens Simon and André Brinkmann: *Autonomic Resource Mangement with Support Vector Maschines In: Grid 2011, Proc. of 2011 IEEE/ACM 12th Int. Conf. on Grid Computing, pp. 157–226–128–147164, 2011*.

Alberto Miranda, Sascha Effert, Yangwook Kang, Ethan Miller, Andre Brinkmann and Toni Cortes
Reliable and Randomized Data Distribution Strategies for Large Scale Storage Systems
Proceedings of High Performance Computing Conference (HiPC), 2011.

Christian Plessl and Marco Platzner
Reconfigurable Embedded Control Systems: From Modelling To Implementation
Hardware Virtualization on Dynamically Reconfigurable Embedded Processors, IGI Global, 2011. accepted for publication

Georg Birkenheuer, Andre Brinkmann, Jürgen Kaiser, Axel Keller, Matthias Keller, Christoph Kleineweber, Christoh Konersmann, Oliver Niehörster, Thorsten Schäfer, Jens Simon and Maximilan Wilhelm
Virtualized HPC: a contradiction in terms?
Software: Practice and Experience, 2011.

Georg Birkenheuer, André Brinkmann, Mikael Högqvist, Alexander Papaspyrou, Bernhard Schott, Dietmar Sommerfeld, Wolfgang Ziegler
Infrastructure Federation Through Virtualized Delegation of Resources and Services
Journal of Grid Computing, vol. 9, no. 3, pp. 355--377, Springer Netherlands, 2011.

Dissertations 2011

Sascha Effert
Verfahren zur redundanten Datenplatzierung in skalierbaren Speichersystemen, 2011

Tobias Schumacher
Performance Modeling and Analysis in High-Performance Reconfigurable Computing, 2011

Organization of Workshops

26th IEEE (MSST2010) Symposium on Massive Storage Systems and Technologies, Denver, USA, Mai 2010

5th IEEE International Conference on Networking, Architecture and Storage (NAS), Macau, Juli 2010

Participation at Fairs

International Supercomputing Conference 2010, Hamburg, Juni 2010
International Supercomputing Conference 2011, Hamburg, Juni 2011

Awards

“Transferpreis OWL 2010” für adaptive Prothesen, Prof. Dr. Marco Platzner, PC^2; Martin Hahn, ixtronics; Michael Winkler, Orthopädietechnik Winkler given by the IHK Ostwestfalen-Lippe

4 Services

4.1 Operated Parallel Computing Systems

In 2010 and 2011 the PC² operated nine high performance computing systems and one parallel file system. Five of the HPC systems were dedicated to specific projects and/or working groups. Four HPC systems were available for all internal and external researchers of the University.

4.1.1 Publicly Available Systems

Name	*Years of Operation*	*Number of Nodes*	*Number of Cores*	*Main Memory*	*Processors*	*Interconnect*	*Machine*
Arminius+	*2010 -*	*60*	*720*	*36 GByte per node*	*Intel X5650 2.67GHz*	*InfiniBand QDR*	*Fujitsu RX200S6*
SMP Compute Server	*2009 -*	*1*	*24*	*128 GByte*	*four Intel X7542 2.67GHz*		*Fujitsu RX600*
High Throughput Cluster (HTC)	*2009 -*	*depends on availability*	*virtual machines with 1 core*	*up to 8 GByte per virtual machine*	*Intel or AMD x86-64bit*	*only for sequential jobs*	*diverse*
Arminius	*2005 - 2010*	*208*	*416*	*4 GByte per node*	*Intel Xeon 3.2GHz, AMD Opteron 2.4GHz*	*InfiniBand SDR*	*Fujitsu hpcLine*

Arminius Cluster

With the financial support of the state North Rhine-Westphalia and the federal republic of Germany, PC² established 2005 the ARMINIUS cluster. In co-operation with Fujitsu-Siemens Computers we designed the system consisting of 200 compute and 8 visualization nodes. The official opening with a ceremonial inauguration was at June, 21st 2005.

Hardware	***Description***
200 dual processor Intel Xeon	400 processors, each node with 3.2 GHz, 1 MByte L2-cache 4 GByte DDR2 main memory 4x InfiniBand PCI-e HCA
8 dual processor AMD Opteron	16 processors, each node with 2.4 GHz, 1 MB L2-cache 12 GByte DDR main memory 4x InfiniBand PCI-e HCA nVidia Quadro FX 4500G PCI-e graphics card
216 port InfiniBand switch fabric	Central switch fabric with 18 switch modules each with 12 ports
7 TByte parallel file system	Accessible from all nodes
1 login node	Used for compiling and starting applications
Stereoscopic rear projection	1.80m x 2.40m screen 2 D-ILA projectors 3D tracking system

Table 1: Hardware specification of the Arminius Cluster

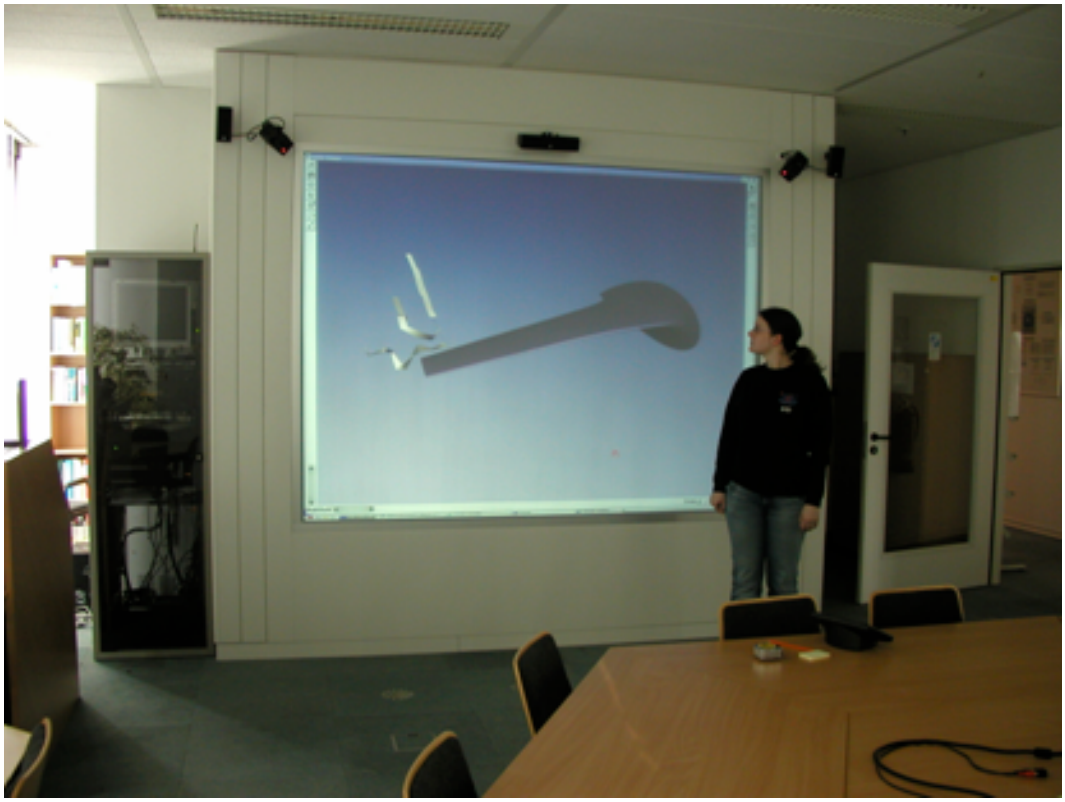


Figure 1: The stereoscopic rear projection of the Arminius cluster

The Arminius cluster with its 416 processors had a peak performance of 2.6 TFlop/s. This compute performance needed about 70k Watt electrical power which led to a nearly equal amount of thermal energy. Our computing center was not able to get that much energy out of the room with the installed air conditioning system. Therefore a special fluid based cooling system was used inside the system. All processors of the compute nodes had special heat sinks, which were connected via a heat exchanger to the cooling system of the building. This technique was able to move 50 to 60 percent of the thermal energy directly out of the room. The rest was cooled with the air conditioner.

We provided standard system software for cluster systems. A Linux operating system with its software development tools was installed. Additionally, some MPI message passing libraries thereunder, three MPI versions optimized for InfiniBand were available. Scientific libraries for numerical applications were provided and the Intel compiler suite optimized for the Intel Xeon processor could be used. The system software environment of the Arminius cluster is shown in the following table:

Software	***Description***
RedHat Advanced Server Release 5	Linux operating system 2.6.9 kernel
GNU Tools	e.g. gcc
Intel compiler	C/C++, Fortran
Scali-MPI-Connect	MPI 1 compliant, fail-over from IB to GbE
MvAPICH	MPICH on VAPI from Ohio State University
MPICH-vmi	MPICH for VMI from NCSA
Intel MKL	Math Kernel Library

Table 1: System Software of the Arminius Cluster

Company	*Components*
Fujitsu-Siemens Computers GmbH	general contractor cluster system
ICT AG	housings, system integration
SilverStorm Technologies	InfinIO 9200 switch fabric 216 ports (max 288 ports)
SilverStorm Technologies Mellanox Inc.	InfiniBand Host Channel adaptor PCI-E IB 4x
nVidia GmbH	graphics cards
Rittal AG	heat exchanger,racks, controlling and management of cooling system
Atotech GmbH	fluid based heat sinks
Intel GmbH	INTEL Xeon EM64T processors HPC software tools
Scali Inc.	MPI Connect
UNITY AG	general contractor visualization equipment
3-Dims GmbH	Integrator of visualization equipment

Table 2: Companies involved in the development of the Arminius Cluster system.

The system was able to sustain 1.978 TFlop/s out of 2.6 TFlop/s peak performance. Based on the Linpack benchmark for supercomputers, the Arminius cluster achieved rank #205 and rank #13 of the German supercomputers in the 25th Top-500 list. The Arminius system was embedded in the German D-Grid and in two worldwide used Grid Computing environments: Globus and UNICORE.
In September 2010, we switched off the ARMINIUS cluster and replaced it by a new system.

Utilization

Figure 2: Arminius utilization 01.01.2010-15.09.2010 (24h per day)

Table 4 depicts the outage dates in 2010 (marked with numbers in Fig.2) the system was not available.

Event	Date	Reason
1	09.03.10	Maintenance
2	11.08.10	Power failure in the whole building
3	15.09.10	System switched off

Table 3: Dates in 2010, the system was not available

Arminius+ Cluster

With the financial support of the state North Rhine-Westphalia and the federal republic of Germany, PC² established 2010 the ARMINIUS+ cluster. It is the successor of the ARMINIUS cluster. The official opening was at October 20th 2010.

Hardware	***Description***
60 Fujitsu RX200S6 Intel X5650	720 processor cores, each node with 2.67 GHz, dual socket, six-core, 36 GByte DDR3 main memory 4x InfiniBand QDR PCI-e HCA
InfiniBand switch	216 port Fabric shared with the ARMINIUS cluster
48 TByte file system	NAS storage shared with all major HPC systems
1 login node	Nodes are used for compiling and starting applications

Table 4: Hardware Specification of the Arminius+ Cluster

The Arminius+ cluster with 720 cores has a peak performance of 7.7 TFlop/s. We provide standard system software for cluster systems. A Linux operating system with its software development tools is installed. Additionally, some MPI message passing libraries optimized for InfiniBand are available. Scientific libraries for numerical applications are provided and the Intel compiler suite optimized for the Intel processor can be used. The system software environment of the Arminius+ cluster is shown in the following table:

Software	***Description***
CentOS 5.6	Linux operating system 2.6.18 kernel
GNU Tools	e.g. gcc
Intel compiler	C/C++, Fortran
MvAPICH	MPICH on VAPI from Ohio State University
Intel MKL	Math Kernel Library

Table 5: System Software of the Arminius+ Cluster

Utilization

Fig. 3 depicts the utilization of the system in 2010. The average load was 41,56%.
Fig. 4 depicts the utilization of the system in 2011. The average load was 75,32%.
Table 7 depicts the outage dates in 2011 (marked with numbers in Fig.4) the system was not accessible

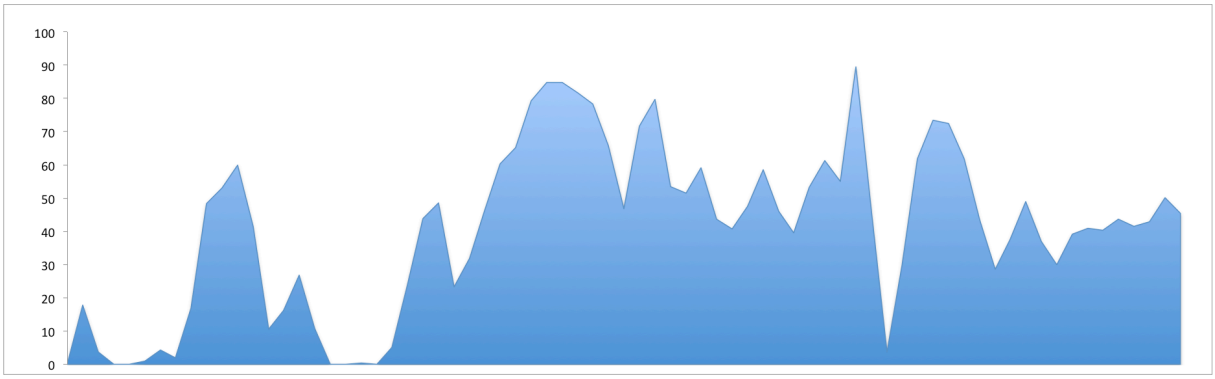


Figure 3: Arminius+ utilization 20.10.2010 – 31.12.2010 (24h per day)

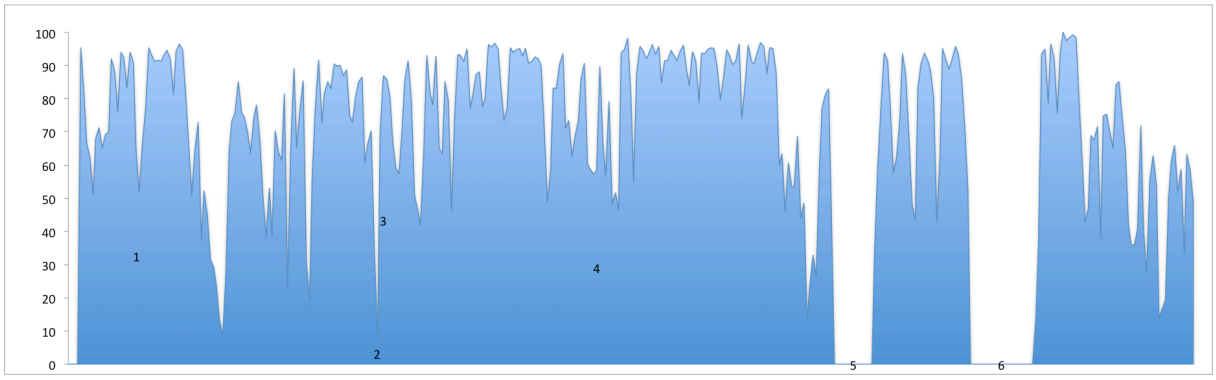


Figure 4: Arminius+ utilization 2011 (24h per day)

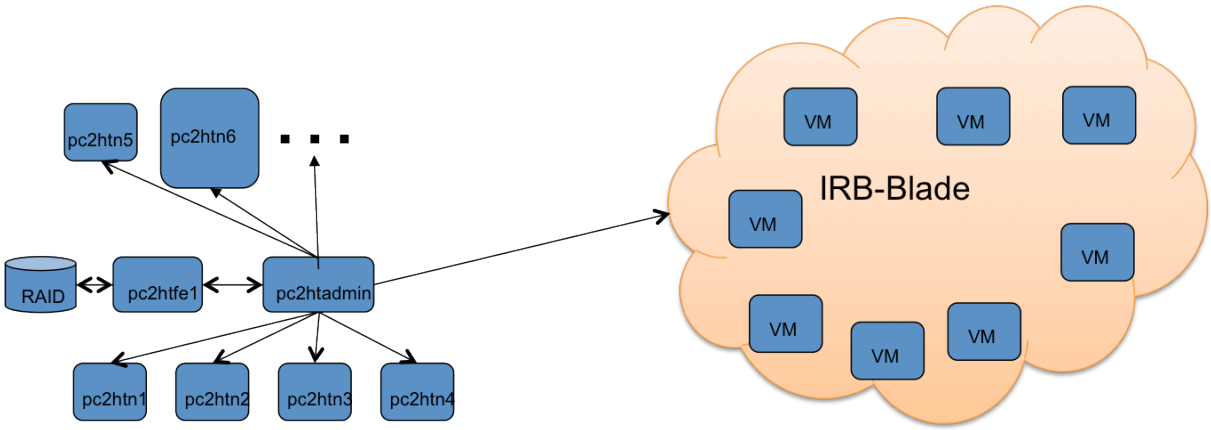
Event	Date	Reason
1	22.01.11 – 24.01.11	Failure in the HA cluster
2	11.04.11	Maintenance
3	13.04.11	Power failure in the whole building
4	21.06.11	Maintenance of the UPS
5	05.09.11 – 19.09.11	System dedicated to a benchmark
6	19.10.11 – 14.11.11	Move to the new building

Table 6: Dates in 2011, the system was not available

SMP Compute Server

Installed	2009
Vendor	Intel, Fujitsu-Siemens
Number of nodes / cores	1 / 24
Node type	4x Intel X7542, 2.67GHz, 6 cores
Node Memory	128 GByte
Operating system	Linux (CentOS)

High Throughput Cluster (HTC)



This system has been established in 2009 and is intended to run sequential jobs with no timeout (i.e. sequential throughput computing) only. The HTC provides access to about 140 CPUs. Most of them are running as virtual machines on a blade system of the computer science faculty.

The HTC comprises:

- **A frontend node**
  - pc2htfe1
  - accessible via ssh from 131.234.
  - 4* Intel Xeon 2.4 GHz
  - 4GB main memory
  - CentOS 5.6
  - Linux-Kernel 2.6.18
- **A server (admin node)**
  - hosting the batch system (Torque)
- **The physical worker nodes**
  - pc2htn*
  - Up to 8GB main memory
  - accessible only via Torque
  - CentOS 5.6
  - Linux-Kernel 2.6.18

- **The virtual worker nodes hosted on a blade system of the computer science faculty**
  - accessible only via Torque
  - 8GB main memory
  - CentOS 5.6
  - Linux-Kernel 2.6.18
  - The master RAID hosting the cluster local homes and the pre-installed software

The following software is installed on the frontend and the worker nodes

- Automake
- Autoconf
- Boost
- DDD
- Emacs
- Gcc
- Java
- Matlab
- Octave
- Perl
- Python
- Valgrind
- Vim

The HTC is only dedicated to members of the University Paderborn.
To be able to use the HT-Cluster one has:

- to have an IMT account and an AFS home
  - AFS home can be activated by using the "IMT Benutzerverwaltung".
- to be member of a special group
  - Send an email to your local support-team to apply for using the HTC.

4.1.2 Dedicated Systems

Name	*Years of Operation*	*Number of Nodes*	*Number of Cores*	*Main Memory*	*Processors*	*Interconnect*	*Machine*
Paderborn HPC Cloud	*2010 -*	*38*	*304*	*16 GByte per node*	*Intel E5506 2.13GHz*	*Gigabit-Ethernet*	*Fujitsu CX1000 CX120S1*
Pling2	*2009 -*	*57*	*456*	*24GByte resp. 48GByte per node*	*Intel E5540 2.53GHz, Intel X5570 2.93GHz*	*InfiniBand SDR*	*Fujitsu RX200S5*
BisGrid	*2008 -*	*8*	*64*	*64GByte per node*	*AMD Opteron 2220, 2.8GHz*	*InfiniBand DDR*	*Supermicro 2041M-T2R*
WinHPC	*2006 - 2011*	*6*	*24*	*8GByte per node*	*Intel 5160 3.0GHz, AMD Opteron 270 2.0GHz*	*InfiniBand DDR*	*diverse*
Opteron Server	*2004 -*	*2*	*8*	*32GByte per node*	*Intel Opteron 2.2GHz*	*InfiniBand SDR*	*Supermicro*

File Systems

Type	*Manufacturer*	*Years of Operation*	*Capacity*	*Protocols*	*Available on*
Network Attached Storage	*Isilon*	*2009 -*	*54TB*	*NFS, CIFS*	*all systems*
Parallel File System	*IBM*	*2007-2009*	*4TB*	*GPFS*	*Arminius*

Hosted Systems

The PC² provides an environment for IT systems operated by other research groups or institutes of the University of Paderborn. For these systems only floor space, electrical power, and cooling is provided. In the years 2010 and 2011 the following systems were hosted by the PC²:

- FPGA-Cluster, Research Group Schaltungstechnik (Dr. Mario Porrmann)
- Network Infrastructure, IMT

Available Software

Software	Purpose	Licence
Abaqus	Finite element analysis	Dedicated
Ansys	3D FEM solvers	Dedicated
Comsol	Multiphysics simulation software environment	Dedicated
CPLEX	Solver for linear programming	Dedicated
FFTW	Library for computing the discrete Fourier transform (DFT)	None
Gaussian	Electronic Structure (g03 and g09)	PC²
Gromacs	Molecular dynamics	None
MKL	Intel Math Kernel Library	PC²
MOE	Molecular operating environment	Dedicated
Matlab	Technical computing	Campus
MPICH	MPICH 1 and 2 for Ethernet	None
MvAPICH	MPICH on VAPI from Ohio State University	None
NAG	Numerical Libraries	NRW
OpenFoam	Finite element analysis	None
OpenMPI	MPI for Ethernet and Infiniband	None
NWChem	High Performance Computational Chemistry	PC²
OpenFoam	Finite element analysis	None
ORCA	Electronic Structure Program Package	None
Scalasca	Performance optimization of parallel programs	None
Siesta	Electronic Simulations	PC²
StarCC	Finite element analysis	Dedicated
Turbomole	Ab initio Electronic Structure Calculations	PC²
VASP	Ab-initio quantum-mechanical molecular dynamics	PC²
Xilinx	FPGA design software	Dedicated

Paderborn HPC Cloud

Hardware	*Description*
38 Fujitsu CX120S1 Intel Xeon Nehalem	304 processor cores, each node with 2.13 GHz, dual socket, quad-core, 16 GByte DDR3 main memory 10Gb Ethernet
48 TByte file system	NAS storage shared with all major HPC systems
1 login node	Nodes are used for compiling and starting applications

The system has a peak performance of 2.6 TFlop/s. It is used as a host for virtual machines, which are running HPC applications.

Physics InfiniBand Cluster (Pling2)

Hardware	*Description*
49 Fujitsu RX200S5 Intel Xeon E5540	392 processor cores, each node with 2.53 GHz, dual socket, quad-core, 24 GByte DDR3 main memory 4x InfiniBand SDR PCI-e HCA
8 Fujitsu RX200S5 Intel Xeon X5570	64 processor cores, each noode with 2.93 GHz, dual socket, quad-core, 12 GByte DDR main memory 4x InfiniBand SDR PCI-e HCA
InfiniBand switch	216 port Fabric shared with the ARMINIUS cluster
48 TByte file system	NAS storage shared with all major HPC systems
1 login node	Nodes are used for compiling and starting applications

BisGrid Cluster

Installed	2008
Vendor	AMD, Fujitsu-Siemens
Number of nodes / cores	8 / 64
Node type	Opteron, 2.8 GHz, 8 cores
Node memory	64 GByte
System memory	512 GByte
Node peak performance	45 GFlop/s
System peak performance	360 GFlop/s
Operating system	Linux

The cluster consists of a frontend system and 8 compute nodes. All nodes are connected via InfiniBand HCAs to a 24 port InfiniBand 4x DDR switch, via Gigabit Ethernet to a control network, and via FibreChanel to a Storage Area Network.The Storage Area Network consists of a switched 4Gbit/s Fibre Channel fabric and 10 TByte disk storage. The parallel file system GPFS is used for high performance disk access.

The system is operated for the German D-Grid projects MosGrid and DGSI. It is available as a D-Grid resource.

Available Software

- Gaussian
- Gromacs
- NWChem
- TurboMole

WinHPC - Paderborner Windows HPC Compute Cluster

Installed	2006
Vendor	FSC
Number of nodes / CPUs	6/ 24
Node type	4x Intel Xeon 5160, 3.0 GHz
Node type	2x AMD Opteron 270, 2.0 GHz
Node Memory	8 GByte
System memory	48 GByte
Node peak performance	96/32 GFlop/s
System peak performance	224GFlop/s
High speed network type	Infiniband
High speed network topology	Switched
Operating system	Windows HPC Server 2008

The WindowsCCS cluster is able to execute 32-bit and 64-bit applications. An MPI version optimized for the highspeed interconnect InfiniBand is available.

Available Software

- Intel Compiler Suite C/C++ / Fortran
- MS MPI
- Matlab
- ANSYS V11.0
- Altera Quartus II 7.1
- Xilinx ISE 9.2i

4-way Opteron Cluster

Installed	2004
Vendor	AMD, Fujitsu-Siemens
Number of nodes / CPUs	2 / 8
Node type	4x Opteron, 2.2 GHz
Node Memory	32 GByte
System memory	64 GByte
Node peak performance	4.6 GFlop/s
System peak performance	18.4 GFlop/s
Operating system	Linux

The nodes are used as frontends of the ARMINIUS+ cluster.

Hosted Systems

The PC² provides an environment for IT systems operated by other research groups or institutes of the University of Paderborn. For these systems only floor space, electrical power, and cooling is provided. In the years 2010 and 2011 the following systems were hosted by the PC²:

- FPGA-Cluster, Research Group Schaltunstechnik (Dr. Mario Porrmann)
- Network Infrastructure, IMT

Available Software

Software	Purpose	Licence
Abaqus	Finite element analysis	Dedicated
Ansys	3D FEM solvers	Dedicated
Comsol	Multiphysics simulation software environment	Dedicated
CPLEX	Solver for linear programming	Dedicated
FFTW	Library for computing the discrete Fourier transform (DFT)	None
Gaussian	Electronic Structure (g03 and g09)	PC²
Gromacs	Molecular dynamics	None
MKL	Intel Math Kernel Library	PC²
MOE	Molecular operating environment	Dedicated
Matlab	Technical computing	Campus
MPICH	MPICH 1 and 2 for Ethernet	None
MvAPICH	MPICH on VAPI from Ohio State University	None
NAG	Numerical Libraries	NRW
OpenFoam	Finite element analysis	None
OpenMPI	MPI for Ethernet and Infiniband	None
NWChem	High Performance Computational Chemistry	PC²
OpenFoam	Finite element analysis	None
ORCA	Electronic Structure Program Package	None
Scalasca	Performance optimization of parallel programs	None
Siesta	Electronic Simulations	PC²
StarCC	Finite element analysis	Dedicated
Turbomole	Ab initio Electronic Structure Calculations	PC²
VASP	Ab-initio quantum-mechanical molecular dynamics	PC²
Xilinx	FPGA design software	Dedicated

4.1.3 System Access

The access to the systems at the PC² is free of charge for all users coming from the academic world e.g. universities or colleges. Users from commercial sites are also welcome but may have to pay a fee for using the systems. Please send an email to the PC² (pc2-info@upb.de).

Access to systems dedicated to specific user groups may be denied depending on the requirements of the owner.

To apply for an account for the PC² systems, one has to fill in small application forms available on our web server. Refer to http://pc2.uni-paderborn.de/become-a-pc2-user

After processing the application, all necessary information will be sent via email within a few days.

The registration information is kept private and will not be disclosed to third parties. It helps us to keep track about the usage of our parallel systems.

Specialist counseling is available for the following fields:

- Compiler
- Debugging
- Grid Computing
- MPI
- Optimization
- Performance Profiling
- System Access
- System-Benchmarking and -Evaluation

For detailed information about how to use our systems please also refer to this URL:
http://pc2.uni-paderborn.de/hpc-systems-services/available-systems/

Please report your problems to: pc2-gurus@upb.de

4.2. Collaborations

4.2.1 Ressourcenverbund – Nordrhein-Westfalen (RV-NRW)

Project coordinator:	Dr. Jens Simon, PC², University of Paderborn
Project Members:	Axel Keller, PC², University of Paderborn

The Ressourcenverbund – Nordrhein-Westfalen (RV-NRW) is a network of university computer centers of the state North Rhine-Westphalia which provides a network of excellence and cooperative resource-usage of high performance compute systems [1]. Targets of this network are:

- Outsourcing of work besides the main focus of each computer center.
- Providing access to short and expensive resources.

Active member organizations of the RV-NRW are:

- RWTH Aachen
- University Köln
- University Paderborn
- University Münster
- University Siegen
- University Dortmund
- University Duisburg-Essen
- Ruhr-University Bochum
- Open University Hagen

In generally, all systems and services of the Ressourcenverbund are available for all scientists of RV-NRW members. The use of the resources is free of charge for this community.

The RV-NRW excellence network provides different kind of services to researchers of universities and institute of the state North Rhine-Westphalia.

Consulting HPC users: The RV-NRW provides a primary point of contact for users for all resources provided within the network. Expert advice will be provided by the appropriate compute center staff responsible for the requested resources. PC² provides all technical

services and user support for its systems. Additionally, courses and material concerning high performance computing are offered to increase the skills and qualifications of the users.

HPC systems and application software:
Several high-performance computer systems are available for the users of the RV-NRW. The *Rechen- und Kommunikationszentrum* of the RWTH Aachen provides a cluster system with 60 nodes, each 2 quad-core processors and a cluster system with 1250 cluster nodes, each with 2 six-core processors, and 376 nodes, each 4 eight-core processors, and a system with 4 TByte shared-memory and 64 processors, each with eight-cores. The *Paderborn Center for Parallel Computing* of the University Paderborn provides the ARMINIUS+ cluster system with 60 nodes, each node with two six-core processors. The University Siegen operates a 128 nodes cluster with two AMD Opteron processors per node.
The following centers are providing resources to RV-NRW, but they are up to now not integrated in the unified user management: The *Zentrum für Informationsverarbeitung* of the University Münster operates an InfiniBand connected cluster system with 20 nodes, each with 2 quad-core processors. The *Zentrum für Angewandte Informatik* of the University Köln provides a cluster system with 817 dual-processor nodes and 16 quad-processor nodes. Finally, 384 blade nodes with 8 cores each are provided by the University Dortmund.
Interested scientists apply for access to the RV-NRW compute resource at their local compute center.

Certificate Registration Authority: The Open University Hagen provides a Public Key Infrastructure (PKI) for an automatically issue of X.509v3 certificates. Members of the RV-NRW are free to use the dedicated certificate-server.
The University Paderborn, PC² is a registration authority for Grid certificates The standard DFN certificates, used for the encryption of e-mails, can not be used for grid services.

Resource Usage
PC² provides about 30 percent of the compute resources of the Arminius cluster to users of universities and institutes of North Rhine-Westphalia. Researchers from University Münster, RWTH Aachen, and Ruhr-Universität Bochum are currently using RV-NRW accounts to access the PC² cluster system.
Further information about the RV-NRW network of excellence is available on the web-pages of the Ressourcenverbund-NRW [1].

References
[1] Ressourcenverbund Nordrhein-Westfalen (in German), http://www.rv-nrw.de

4.3. Teaching

4.3.1 Theses and Lectures in PC²

Lectures

- Operating Systems
  (WS10/11 – Jun.-Prof. Dr.-Ing. André Brinkmann)
- Compact Course on Theoretical Aspects of Storage Systems Research
  (WS09/10 – Jun.-Prof. Dr.-Ing. André Brinkmann)
- Architektur Paralleler Rechnersysteme
  (WS10/11, WS11/12 – Dr. Jens Simon)
- Software-Praktikum für Ingenieur-Informatiker
  (WS09/10 – Jun.-Prof. Dr.-Ing. André Brinkmann)
- Storage Systems
  (SS10, SS11 - Jun.-Prof. Dr.-Ing. André Brinkmann)
- Hardware/Software Codesign
  (SS10, SS11 – Dr. Christian Plessl)

Project Groups

- Development of a flexible high-performance File System
  (2010/2011 – Jun.-Prof. Dr.-Ing. André Brinkmann, Dirk Meister, Matthias Grawinkel)
- Bioinformatics Custom Computers
  (2011/2012 – Jun.-Prof. Dr. Christian Plessl, Tobias Kenter, Heiner Giefers)

Master´s Theses

- Krieger, Alexander: Automated provisioning of virtual machines in a cloud environment, 2011
- Kaiser, Jürgen: Verteilte fehlertolerante inline Deduplizierung, 2011
- Wiersema, Tobias: Scheduling Support for Heterogeneous Hardware Accelarators under Linux, 2010
- Schäfer, Thorsten: Energieeffiziente fehlerkorrigierende Codes in festplattenbasierten Datenarchiven, 2010
- Bolte, Matthias: Non-intrusive Virtualization Management using libvirt, 2010
- Konersmann, Christoph: Checkpointing von parallelen Anwendungen in virtualisierten Umgebungen, 2010

Bachelor´s Theses

- Kinscher, Johnannes: Parallelisierung eines Konsistenzalgorithmus zur Szenarioberechnung, 2011
- Hartung, Tim: Leistungssteigerung von EXT2 Dateisystemen durch Separation der Metadaten, 2011
- Lauhoff, Marcel: Evaluation of Java-based Filesystems, 2011
- Lipp, Matthias: Cost Analysis of Backup Approaches, 2011
- Moors, Sebastian: Metadatenmanagement zur energieoptimierten Langzeitarchivierung, 2010
- Strothmann, Tim: Management virtueller Umgebungen, 2010
- Kleineweber, Christoph: Regelbasierte Platzierung virtueller Maschinen in Clouds, 2010

PC² Colloquium 2010

- Prof. Dr.-Ing. Weinhardt, Markus: PACT XPP-III: Architektur und Programmierung
- Mutke, Ernst M.: Putting Personality Into High Performance Computing

PC² Colloquium 2011

- Dr. Krüger, Jens: Scientific Comuputing

PhD Theses

- **Dr. Sascha Effert**
  Verfahren zur redundanten Datenplatzierung in skalierbaren Speichersystemen, 2011

- **Dr. Tobias Schumacher**
  Performance Modeling and Analysis in High-Performance Reconfigurable Computing, 2011

4.3.2 PhD at PC²

Dr. Sascha Effert: "Verfahren zur redundanten Datenplatzierung in skalierbaren Speichersystemen**"**

Abstract
Modern data centers are faced with a rapidly growing amount of data which they have to keep highly available with increasing performance. Therefore, they need storage systems, which grow with the demands. Storage networks are more and more used in this area. In such systems, data servers create a virtual storage on top of a number of physical hard disks. Therefore, the load of the virtual storage has to be distributed in a way that the physical disks are used in an optimal way. It is also important to store all data with redundancies to be able to compensate broken hardware. The algorithms used for data distribution have to solve all these demands. Pseudo randomized hash functions have a big impact in this area.

In my dissertation I describe different storage systems. I take a closer look at storage networks with data servers using directly attached hard disks. For these I show how they scale using different kinds of data distribution. Unfortunately, none of the described storage systems solves all demands.

As a solution I introduce Redundant Share which solves all demands. Using Redundant Share it is possible to distribute an arbitrary number of copies of each piece of data using each hard disk in an optimal way according its capacity. Moreover, adding a new physical hard disk ends up in a bounded effort. I perorate with this by measuring an implementation of Redundant Share and by comparing the results with other distribution algorithms.

Dr. Tobias Schuhmacher: “Performance Modeling and Analysis in High-Performance Reconfigurable Computing”

Abstract

Reconfigurable computing has received a high level of attention during the last years. Scientists presented accelerators for different algorithm classes gaining speedups of several orders of magnitude. Major supercomputer vendors came out with high-performance computers that tightly connect reconfigurable devices to the CPUs and/or to the memory subsystem. One of the major focuses of recent research is put on the programmability of these reconfigurable high-performance computers. Despite great research results in this topic, there are still several challenges which make the development process of reconfigurable accelerators a time consuming and error-prone process.

One of the main issues in this area is the question whether reconfigurable computing is even able to generate a benefit for specific applications. Since the design of reconfigurable accelerators typically is a very time consuming process, it is mandatory to estimate the potential of this technology before actually implementing the accelerator. For this purpose, modeling techniques are required. Existing modeling approaches are often restricted to a static architecture model and provide methods for specifying algorithms to be executed on those specific architecture models. These techniques are not well-suited when considering reconfigurable computing, since in these cases the concrete architecture is typically generated explicitly for the specific algorithms.

In order to analyze the performance potential of the generated accelerator model, a deep knowledge of the underlying architecture is required. This includes especially the time necessary for data transfers between CPU and accelerator as well as the time needed for memory access. Many tools exist for measuring low level performance values on commodity CPUs, but no corresponding tools are available to measure such values for reconfigurable hardware.

The design and implementation of reconfigurable accelerators lead to the demand for a development framework that supports the designer in implementing and testing the modeled design. Simulating a complete design typically requires a large amount of time, so the development framework should support in-system performance monitoring. Another key issue is the portability of the framework and the resulting accelerators.

This thesis introduces a novel approach to meet these requirements. It introduces a modelin technique which supports algorithms targeted at commodity CPUs as well as reconfigurable accelerators. A key point of the modeling technique is that it does not assume a static architecture model, but allows for specifying the architecture model along with the execution model of the algorithms to be implemented. Additionally, the modeling approach does not only focus on the execution time of arithmetic operations performed, but also on the time needed for data transfers.

The modeling approach is supported by the IMORC architectural template which eases the implementation of the modeled accelerator. The architectural template assumes accelerators to be implemented as a set of communicating cores. Each core may reside in its own clock domain and communicate to others using an on-chip network. A key feature is that the network allows control structures and datapaths to be implemented completely independently from each other. Integrated performance counters support the debugging and the in-system performance analysis of the final accelerator.
In order to further support the modeling phase, an architecture characterization framework based on the architectural template is introduced. This framework allows to measure the communication bandwidth between CPU and reconfigurable hardware as well as between reconfigurable hardware and different kinds of memory in detail. It supports different kinds of communication schemes and can also generate contention on the target memory by accessing it concurrently using multiple cores.
The introduced approach is finally evaluated by demonstrating three case studies out of different problem domains. These case studies show that the presented approach greatly helps in analyzing algorithms concerning their acceleration potential on reconfigurable hardware and in implementing and optimizing the final accelerators. The accelerators are implemented using only a small amount of hand written VHDL code. Most functionalities are realized using the features of the IMORC architectural template. The integrated load sensors helped significantly to identify bugs and performance bottlenecks during the design phase. Those would have been hard to find using only simulation techniques.

4.3.3 Project Group: Development of a flexible high-performance File System

Project coordinator	Jun.-Prof. Dr. André Brinkmann, PC², University of Paderborn
Project members	Dirk Meister, PC², University of Paderborn Matthias Grawinkel, PC², University of Paderborn

Research, companies, and governments require to store and process huge amounts of data. Today's file systems are often spread to multiple parallel storage and metadata servers to cope with these demands. These file systems can store multiple petabytes of data and serve thousands of clients in parallel.
The novel pNFS (NFSv4.1) protocol allows a standardized access to a clustered file system and its server-side implementation yields a broad design space that affects the scalability, extendibility and overall performance of the file system.

Goals
Most file systems are tailored for specific workloads, while the goal of this project is to exploit standard protocols and modern technologies to create a very flexible and configurable distributed file system. Therefore, a standard compliant (distributed) pNFS server should be developed that could be used as a building block for scalable, efficient storage solutions. This includes:

- Implementation of a scalable, distributed pNFS Server
- Exploitation of the given protocols to dynamically add new features
- Addition of new storage layouts that support novel features like deduplication
- Definition and implementation of Server-to-Server protocols
- Performance evaluations

References
[1] Web-page: http://pc2.uni-paderborn.de/teaching/lectures/project-group-flexible-distributed-file-system/

4.3.4 Project Group: Bioinformatics Custom Computers

Project coordinator	Jun.-Prof. Dr. Christian Plessl, University of Paderborn
Project members	Tobias Kenter, PC², University of Paderborn Heiner Giefers, PC², University of Paderborn

Modern molecular and systems biology analyze genome information to obtain a better understanding of organisms. The resulting data is used for example for studying the function of the genes and their interaction, understanding the metabolism of organisms, finding the causes and cures for diseases, etc. The genome information is stored in the DNA of the organism and can be decoded using a DNA sequencer, which assembles fragmentary DNA pieces.

In recent years DNA sequencing technology has advanced rapidly and new high-throughput DNA sequencers are revolutionizing biomedical research. Starting in 2005, a variety of novel massively parallel sequencing instruments such as the Roche/454, the Life Technologies SOLiD, and the Illumina platforms have been introduced which are sufficiently fast to sequence complete human and model organism genomes. Over the last years the DNA sequencing capacity of next-generation sequencers has been doubling every six months while the cost of sequencing a genome is decreasing. The flip side of this progress is that the vast amount of data created by these high-throughput sequencers leads to serious challenges when processing and archiving this data. Hence, it is foreseeable that in the close future compute performance, instead of sequencing, will become the bottleneck in advancing genome science.

One promising approach to accelerate processing of genomic data is the use of novel parallel computer architectures. In this project group, we will investigate the use of reconfigurable computers which provide field-programmable gate arrays (FPGAs) as programmable co-processors to offload computation. FPGAs have shown a high potential for accelerating the critical operations in many bioinformatics workloads by several orders of magnitude due to their support for massively parallel computation and explicit control over fine-grained parallelism. While previous generations of reconfigurable computers were difficult to program since the developer had to manually translate algorithms to hardware circuits, recent machines are programmed with high-level languages.

Goals

The aim of the project group is to design and program custom computing platforms for specific bioinformatics algorithms. The project group will have access to two of the hottest and most powerful reconfigurable computers that are currently available: the Convey HC-1 Hybrid Core Computer and the Maxeler MaxWorkstation for developing accelerators for bioinformatics problems like "short read mapping" or "sequence alignment". Our goal is to outperform the fastest currently existing implementations for the selected algorithms. The performance of the developed implementations will be validated with genuine genomic data provided by our cooperation partner CeBiTec Bielefeld, which is one of the leading bioinformatics centers in Germany.

References

[1] Web-page: http://www.cs.uni-paderborn.de/fachgebiete/computer-engineering-group/teaching/ws1112/pg_bio.html

5 Research Projects

5.1 Computer Architecture

5.1.1 Lonestar: An Energy-Aware Disk Based Long-Term Archival Storage System.

Project coordinator	Jun.-Prof. Dr. André Brinkmann, PC², University of Paderborn
Project members	Dirk Meister, PC², University of Paderborn Matthias Grawinkel, PC², University of Paderborn Ivan Popov, PC², University of Paderborn Jürgen Kaiser, PC², University of Paderborn Yan Gao, PC², University of Paderborn
Work supported by	BMWi, christmann Medien- und Informationstechnik GmbH, Scalus

General Problem Description
Modern data centers are constantly faced with increasing demands for storage. While in the past years the main problems were located in the exponentially increasing single-disk size compared to the linearly growing IO throughput, our focus now moves to resource efficiency.

The PC^2 is working on resource efficiency in form of data deduplication and energy-efficient archival storage. Data deduplication is a technique to reduce overall storage capacity consumption, and, thus, the number of needed disks, by exploiting data redundancy. The increasing storage demands forced researchers to design distributed deduplication environments, which face the new problem of being scalable while concurrently maintaining a high redundancy detection ratio. Most systems focus on high performance and neglect the deduplication ratio. The PC^2 chose a different approach and developed a distributed deduplication system with exact deduplication and good scaling properties based on its dedupv1 approach.

Another approach for resource efficiency lies in energy-efficient archival storage. In recent years, disk properties improved in a way that they became a reasonable alternative to tape if combined with a RAID system. However, traditional RAID techniques are rather optimized for IO throughput and reliability than for low energy consumption. New RAID schemes and data placement strategies are necessary to reach this goal.

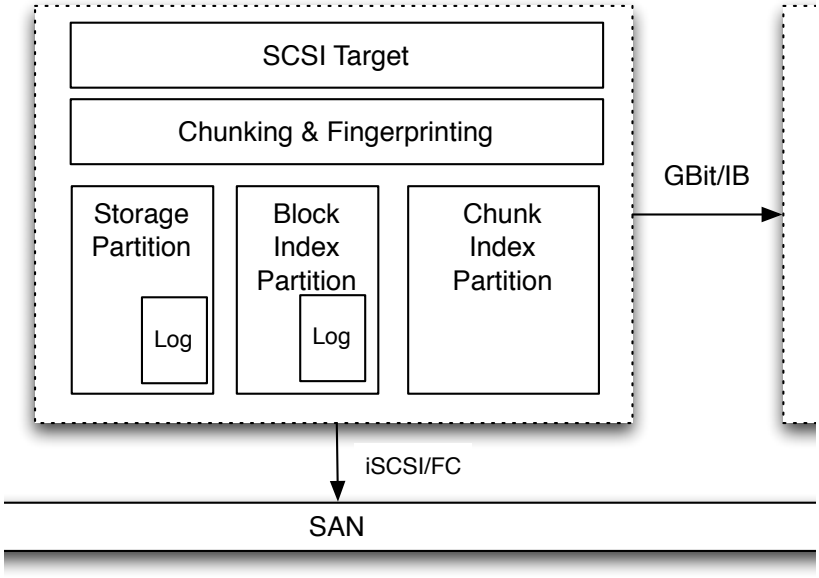
Storage systems tend to use clustered architectures designed to efficiently store and process such big amounts of digital information. With increasing frequency, these systems have Quality of Service (QoS) requirements. This introduces new challenges in storage system development, like load-balancing and data distribution. A variety of randomized solutions handling data placement issues have been proposed and utilized. However to the best of our knowledge, there has not yet been a structured analysis of the influence of pseudo random number generators (PRNGs) on the data distribution.

Problem details and work done

The data deduplication research done by the PC^2 focuses on the throughput and scalability aspects. To enable this research, the PC^2 developed the dedupv1 block-based deduplication system. The architecture of the original single-node variant is presented in Figure 1.

Naive deduplication systems store the fingerprint information necessary to perform the redundancy detection either on disk or in memory. If the information is stored on disk, the throughput is limited to around 20 MB/s, even if using a large disk array, because the number of random IO operations per disk is limited to less than 200 per second. If the information is stored in main memory, the scale of deduplication is limited to a few terabytes because main memory has a prohibitive price for sizes of more than 64 GB. The PC^2 research has overcome this dilemma by combining two directions: Solid State Disk and Cluster Storage.

Solid State Disks (SSDs) are new storage systems based on flash memory but provide a disk-like interface. These SSDs have no mechanical moveable parts and provide a much larger random IO performance than disk, but they are limited in capacity and lifetime. The dedupv1 storage system designed at the PC^2 stores the fingerprint information on a series of Solid State Disks to increase the throughput of deduplication systems [5].



While the problem of big data structures can be solved via SSDs, the scalability properties of single node deduplication systems remain poor. To solve this, the PC^2 extended the dedupv1 design to scale horizontally to multiple nodes as a clustered deduplication system. Unlike other systems, the clustered dedupv1 provides global deduplication. This means that the system detects all redundancies regardless of which cluster nodes are used or on which cluster node the redundant data has been stored on before. While normal cluster storage has been a research topic for a few years, cluster deduplication provides different tradeoffs and different research issues.

New topics in the area of data deduplication systems that came up during the report's time span are aging-properties and advanced garbage collection approaches for distributed deduplication systems. Furthermore, we cooperate with multiple HPC and supercomputing providers, such as the Barcelona Supercomputing Center (BSC) and the DKRZ in Hamburg, to analyze in depth if and how deduplication systems might be worthwhile for HPC (archival) storage. This work builds on the foundations of previous studies [1].

The PC^2 opened the field of energy-efficient archival storages with the newly developed LoneStar storage system. In cooperation with the University of Bielefeld and christmann Medien- und Informationtechnik GmbH, new hardware and software architectures were combined to build an energy-efficient archival system whose economic efficiency can compete with tape while providing a more flexible system.

The PC^2 developed the software architecture of the archival storage. We provide modern cloud storage semantics and use concepts known from parallel file systems where the metadata of the stored objects is held separately from their content. A centralized, highly available metadata server is responsible for all operations on the storage system. A

client's request for an object is sent to the metadata server and is then redirected to the responsible storage server. This architecture centralizes the decisive power on the metadata server and provides the foundation for sophisticated data placement strategies and measures to reduce the overall energy-consumption of the system.

The storage server is a custom server architecture that houses 192 2,5" disk drives in a single 3U enclosure. The hardware was developed by our project partners and was designed to be as energy- and resource-efficient as possible. To manage the 192 disk drives, a new backplane architecture was developed, which integrates a microcontroller network and provides means to shut down parts of the server. An interface of the operating system yields mechanisms to dynamically power only the required parts of the server. The very high storage density and the choice of components resulted in an architecture that is both, energy- and resource-efficient.

LoneStar provides a multi-level reliability concept that checks the integrity of stored data and provides means to recover from media errors, full disk and server failures. The metadata server stores objects' hashes and can spread data to multiple storage servers to become resiliant against server failures. To exploit the unique characteristics of our hardware, we developed a new RAID scheme that intertwines multiple simple RAID schemes into a multidimensional scheme that spans all disks [2]. In contrast to conventional RAID systems, we do not spread contiguous data to multiple disks but sequentially write objects to single disks. We optimize four systems with "write once, read sometimes" characteristics and require only a single spinning disk for reads by moving the work overhead to writes. Here, 4 disks are required to update the corresponding parities. On each disk, we use intra-disk redundancy encoding to detect and recover silent data corruption, which improves the system's reliability and reduces the frequency of regular disk scrubbing runs [3,4].

Many load-balancing strategies for storage systems are based on randomized data distribution strategies, especially in the context of balls-into-bins games. Nevertheless, these randomized strategies never completely balance the load from a short-term perspective. The PC^2 investigated the impact of different load-balancing strategies and the impact of using different pseudo-random number generators (PRNG) from a practical perspective.

To reach these goals, the Consistent Hashing distribution strategy was considered as a combination of two consecutive phases: distribution of bins and distribution of balls. PRNGs were analyzed in regards to their efficiency, both phase independently and in terms of their overall behavior. The result of this analysis helps to choose a PRNG according to the quality of the load distribution and the performance. Additionally, we explored PRNGs for different data placement schemes. We investigated the influence of

the distribution strategies on the generators and tried to identify the correlations between PRNG internal algorithm types and their properties [6].

Another approach to improve QoS support for storage systems is to develop and integrate a domain-specific language based on the pNFS protocol, which allows the pNFS metadata server to describe nearly arbitrary layouts and to send these layouts to the clients. The language has to be secure, easily interpretable by the client computers, and should support heterogeneous CPU architectures. Work in this direction has recently been started.

Resource Usage

Storage research has different requirements for system environments than traditional HPC and related research. The researcher, therefore, used additional hardware consisting of commodity servers but with larger internal disk storage, access to remote storage via Fiber Channel, and partly containing an internal SSD. This storage research cluster is used on a daily basis in close cooperation with the Cloud infrastructure efforts of the PC^2.

References

[1] Meister, D. and Brinkmann, A.: Multi-Level Deduplication in a Backup Scenario, Proceedings of the Israeli Experimental Systems Conference (SYSTOR), Haifa, May, 2009

[2] Grawinkel, M.; Pargmann, M.; Dömer, H. and Brinkmann, A.: Lonestar: An Energy-Aware Disk Based Long-Term Archival Storage System, Proceedings of the 17th IEEE International Conference on Parallel and Distributed Systems (ICPADS), Tainan, December, 2011

[3] Grawinkel, M.; Schäfer, T.; Brinkmann, A.; Hagemeyer, J. and Porrmann, M.: Evaluation of Applied Intra-Disk Redundancy Schemes to Improve Single Disk Reliability, Proceedings of the 19th Annual Meeting of the IEEE International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS), Singapore, July, 2011

[4] Gao, Y.; Meister, D. and Brinkmann, A.: Reliability Analysis of Declustered-Parity RAID 6 with Disk Scrubbing and Considering Irrecoverable Read Errors,

[5] Proceedings of the 5th IEEE International Conference on Networking, Architecture and Storage (NAS), Macau, 2010

[6] Meister, D. and Brinkmann; A.: dedupv1: Improving Deduplication Throughput using Solid State Drives (SSD), Proceedings of the 26th IEEE Symposium on Mass Stroage Systems and Technologies (MSST), Incline Village (NV), May 2010

[7] Popov I.; Brinkmann A. and Friedetzky T.: On the influence of PRNGs on data distribution, Proceedings of 20th Euromicro International Conference on Parallel Distributed and Network-Based Computing (PDP), Garching, Germany, February 2012

[8] Lensing P.; Meister, D. and Brinkmann, A.: hashFS: Applying Hashing to Optimize File Systems for Small File Reads, Proceedings of the 6th IEEE International Workshop on Storage Network Architecture and Parallel I/Os (SNAPI), Incline Village (NV), May 2010

5.1.2 IMORC: An Infrastructure and Architectural Template for Performance Monitoring and Optimization of Reconfigurable Accelerators

Project coordinator	Prof. Dr. Marco Platzner, University of Paderborn
Project members	Tobias Schumacher, PC², University of Paderborn

General Problem Description

While research has demonstrated the potential of FPGAs as acceleration technology for decades, the creation of accelerators for realistic workloads by the lack of commercially available systems, appropriate design methods and corresponding implementation techniques.

First work in this area usually used workstations equipped with PCI or PCIe attached FPGA accelerator boards. In recent years, however, major supercomputer vendors started providing servers with integrated, reconfigurable accelerators that are tightly connected to the systems high-performance interconnect. More recently, FPGA modules that fit into processor sockets have been introduced and provide a fairly standardized way of integrating hardware accelerators into mainstream computing systems.

However, designing an accelerator and optimizing its performance still remains a difficult and time consuming task requiring significant hardware design expertise. The first question that arises when considering reconfigurable hardware for accelerating applications is the performance potential provided by the technology in this special case. While identifying the most compute intense parts of an application is an easy task using standard benchmarking and profiling applications, the acceleration potential for these parts is still unknown. The first part of our work in this project is the introduction of a novel modeling approach to estimate an accelerator's performance before designing or implementing it. The model can then be refined step-by-step to specify the implementation of the final accelerator.

For assisting with the implementation of the accelerator, we developed the IMORC infrastructure and architectural template. The architectural template assumes accelerators to be composed of different cores, which communicate via an on-chip interconnect. A tool chain including a ruby-based code generator thereby assists the developer in generating the cores and the interconnect.

Problem details and work done

The proposed modeling approach consists of two major parts: an architecture model and an execution model. The purpose of the architecture model is to describe the underlying system, whereas the execution model is used for describing the application to be

implemented. While typical, traditional modeling approaches are based on a fixed architecture model, the IMORC modeling approach accounts for the fact that the architecture in reconfigurable hardware is not fixed but has to be implemented for a specific application. Therefore in IMORC, the architecture model and the execution model are developed and refined synchronously.

The architecture model's underlying IMORC is a network of communicating cores. A core typically consists of some local storage, such as embedded memory blocks or registers, and an execution unit. Cores can communicate with other cores using an on-chip network. For accessing the network, each core provides an arbitrary number of communication ports, which are either master or slave ports. The network connects master to slave ports using links. Communication can only be initiated by master ports by sending a read or write request to the slave port. For write requests, the master has to send data corresponding to the request to the slave; for read requests, the slave has to reply with a corresponding data packet. Master ports may connect to multiple slave ports, in which case addressing is required to select the target slave port of a communication request. Conversely, slave ports may connect to multiple master ports through an arbiter.

Figure 1 shows the sample diagram of a typical IMORC compute core. The core consists of a slave communication port, which is connected to a set of registers and to a block of local memory. Two IMORC master communication ports are used in this case for sending communication requests to other cores. A request controller for each of these ports is responsible for generating the appropriate sequence of read and write requests. These request controllers read the current job's parameters from the register block. An execution unit is also connected to the communication ports and performs the actual data processing on data received from communication partners. The execution unit may also access data stored in the registers or read and write data from and to the local memory. Shared memory is modeled as a special kind of core, which only consists of on IMORC slave port and a large amount of local memory.

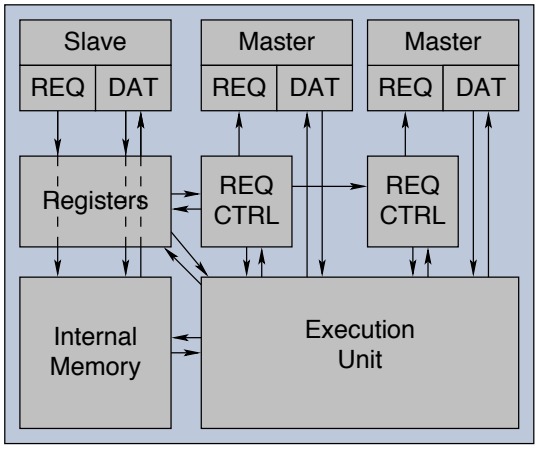


Figure 1: Diagram of a sample IMORC compute core

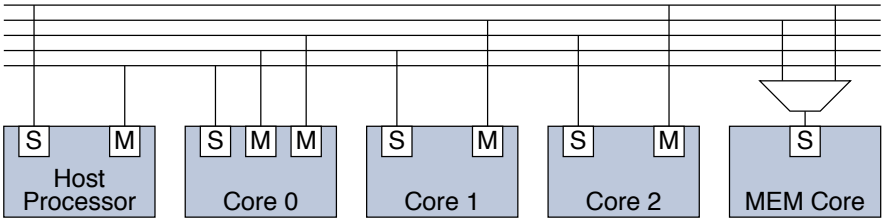


Figure 2: shows an IMORC architecture diagram of an exemplary accelerator.

The accelerator consists of a host processor core, a shared memory, and three compute cores. Each communication bus, represented by the horizontal lines, is connected to exactly one IMORC master port and to one or multiple slave ports. A slave port can be connected to multiple busses by using an arbiter, as shown for the memory core.

The intention of the execution model is to specify the application to be implemented and to support the definition of the architecture model. While the architecture model of IMORC comprises a set of cores that are connected to some kind of network, the execution model has to specify the actual behavior of these cores and their communication pattern. This model consists of a set of tasks, which are able to communicate to each other. Tasks are composed of a number of operations, which are classified into three distinct groups:

- Incoming communication operations so the task has to process incoming messages,
- outgoing communication operations so the task sends messages to other tasks and eventually has to wait for a response, and
- local operations, such as the addition of two values.

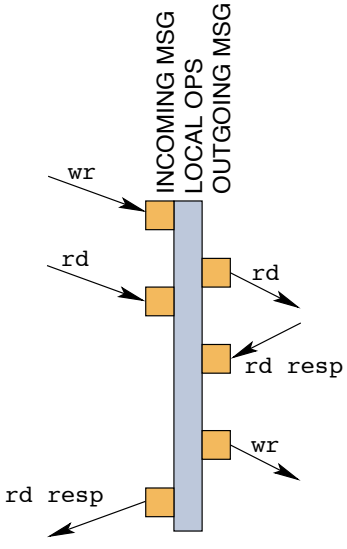


Figure 3: Sample task

Figure 2 shows an example of such a task. The tall box in the middle represents local operations performed by the task. The smaller boxes on the left and right represent communication points with other tasks. Tasks can communicate with other tasks using messages. Messages can be read (rd) or written (wr). The first kind is used for requesting data from another task, the other one for sending data to another task. Read requests need to be followed by a response message (rd resp). Tasks can be modeled at different levels of abstraction, depending on the actual requirements. On a rather abstract level, the tasks' computations may be described by a pseudo code or a code segment in a high-level language. On a fairly detailed level, the computations may be expressed as a sequence of micro operations or RTL codes. Tasks may directly operate on local memory. Shared memory accessed by multiple tasks is modeled as a separate task that communicates with the tasks accessing the memory.

Starting with a very coarse grained task graph and architecture model, where the architecture model only contains the available CPUs, memories, and reconfigurable devices and the task graph consists of one large task representing the complete application, the model is now refined stepwise. The initial task graph is split into multiple communicating tasks that are mapped to the available resources. After this initial mapping, a first performance estimation can be made based on the amount of data that has to be transferred and the bandwidth available between the different resources. Then, the task graph can be further refined to specify the cores required in the accelerator. When the task graph and the architecture model are refined to a reasonable level, a final performance estimation can be made and the implementation can be started.

For supporting the implementation step, we developed the IMORC architectural template. The architectural template assumes accelerators to consist of several communicating cores, as described by the architecture model. Cores communicate using master and slave communication ports, which are connected by links. A port consists of three channels: REQ for transmitting communication requests from the master to the slave, M2S for transmitting data from the master to the slave, and S2M for data transfers from the slave to the master. The packets, which are transmitted on the REQ channel, consist of the start address of the data transfer, the amount of data that has to be transferred, and information if the data has to be read or written. Figure 4 shows the block diagram of an IMORC link. Each of the three channels is buffered in an asynchronous FIFO. Through this buffering, the control and the data path can be implemented independently from each other. The two cores may also operate on data of different width, in which case a bit width conversion module is inserted. Additionally, load sensors are connected to the FIFOs for monitoring the runtime behavior of the application. Multiple master ports may be connected to a single slave core, in which case an arbiter is inserted between the slave cores and the FIFOs. This way, a fast slave core can serve multiple slower master cores.

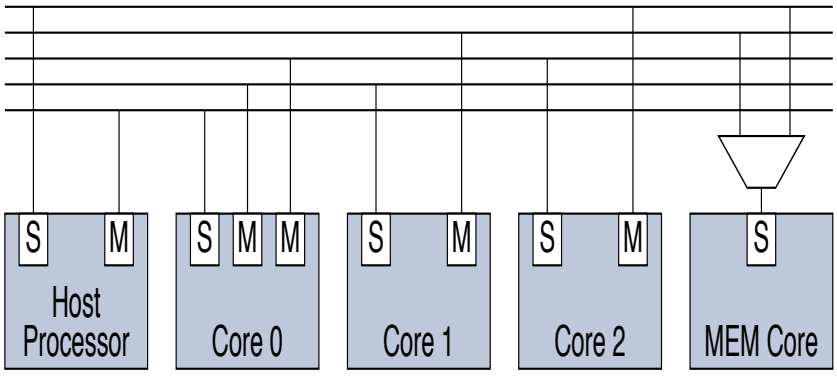


Figure 3: Block Diagram of an IMORC link

In addition to the communication infrastructure, IMORC provides a Ruby-based code generator framework. This framework allows an easy instantiation of compute cores, the generation of the communication infrastructure, and the generation of different communication request generators.
We have shown the potential of this approach with different case studies. In [1] we present the implementation of an accelerator for the k-th nearest neighbor thinning problem. The accelerator was implemented using the IMORC development flow and showed speedups of up to factor 44 over a standard Opteron processor. In [2] we present an accelerator for a 3D image compositing algorithm. Even though the application is very communication-bound, we achieved speedups of up to 2.1, which can be an important improvement when considering animations.

Resource Usage

Development, simulation, and synthesis were mainly performed on several fast server systems. For generating different configurations of an accelerator, the Arminius cluster and the Windows HPC cluster were used occasionally. The evaluation of the architectural template was performed on the XtremeData XD1000 reconfigurable workstation.

References

[1] Schumacher, T.; Plessl, C. and Platzner, M.: An Accelerator for k- th Nearest Neighbor Thinning Based on the IMORC Infrastructure. In Proc. Int. Conf. on Field Programmable Logic and Applications (FPL), pages 338–344. IEEE, September 2009.

[2] Schumacher, T.; Süß, T.; Plessl, C. and Platzner, M.: FPGA Acceleration of Communication-bound Streaming Applications: Architecture Modeling and a 3D Image Compositing Case Study. International Journal of Reconfigurable Computing (IJRC), vol. 2011, 2011. Article ID 760954.

5.1.3 MM-RPU: Multi Modal Reconfigurable Processing Unit

Project coordinator	Prof. Dr. Marco Platzner, University of Paderborn
Project members	Jun.-Prof. Christian Plessl, PC^2, University of Paderborn
	Tobias Kenter, PC^2, University of Paderborn
Supported by:	Intel Microprocessor Technology Lab, Braunschweig

General Problem Description

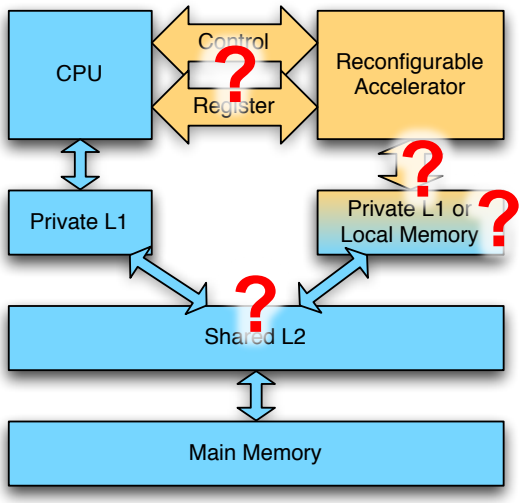
There is an ongoing demand for increased single thread performance in high performance and general purpose computing. One way to serve this demand is the utilization of reconfigurable hardware, like FPGAs. Due to limited hardware resources or a large engineering effort, it is often not desirable to execute the entire application on reconfigurable hardware. Therefore, reconfigurable hardware is commonly combined with a general purpose processor into a system where the reconfigurable part works on the more compute intense parts of the application and the general purpose processor handles the more control flow oriented parts.

A widely researched way to use FPGAs are autonomous or semi-autonomous accelerators that implement the compute intense kernels of an application in the reconfigurable hardware. Another field of research to increase performance is that of extending instruction sets by workload-specific custom instructions. In the embedded systems area, projects work on instruction set extension with reconfigurable hardware, whereas in the high performance area, up to now, new instructions are introduced only as fixed function circuits.

The goal of this research project is to propose and investigate a reconfigurable accelerator that can operate as both, an autonomous accelerator for whole kernels and a functional unit executing custom instructions. The challenges to be tackled are the architectural integration of this accelerator, the design space exploration of its architecture parameters, and the performance estimation for a wide spectrum of applications.

Problem details and work done

The project seeks to combine the concepts of custom instructions and autonomous accelerators. Custom instructions are designed to replace a series of instructions from the existing instruction set by one new instruction. In this way, code size and execution times can be reduced. Custom instructions can make use of instruction



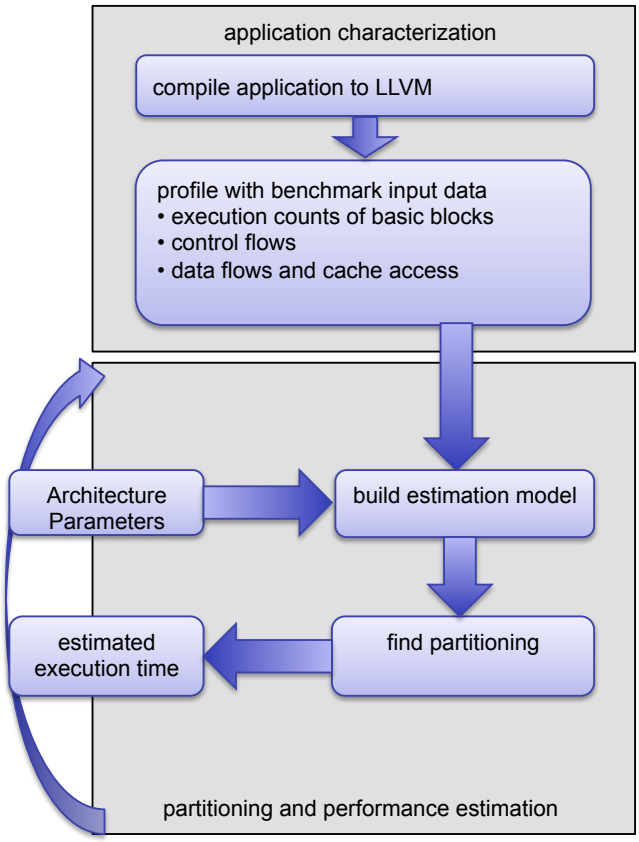
level parallelism by executing several independent operations concurrently and can also exploit bit level parallelism where custom circuits are created. Custom instructions require integration into the CPU that allows for a very low latency communication. They typically work on data from the CPUs register file. The control flow is usually handled entirely by the CPU. Custom instructions typically lead to moderate speedups, maybe with the exception of custom data path instructions using bit level parallelism. On the other hand, some forms of custom instructions can be applied to virtually any application, and there are approaches for automated identification and integration of custom instructions.

Autonomous accelerators can vary greatly in the degree of autonomy, complexity, and architectural coupling. They have in common that the reconfigurable hardware handles at least a part of the control flow of a program. This starts with simple loop accelerators, where the hardware has to maintain an iteration counter of the loop or check for some termination criterion and possibly also compute a stride of memory locations and lead up to designs where the accelerator handles very complex tasks and refers to the CPU for example only for operating system services. Depending on the granularity with which the application is partitioned between the accelerator and the CPU and on the involved datasets, the requirements on latency and bandwidth for the interconnect between the two components vary. For a good flexibility of the reconfigurable hardware, it is desirable that it can be integrated into the CPU's address space, work on the same physical memory, and possibly share a part of the cache hierarchy with the CPU.

An illustration of the proposed architecture is given in Figure 1. Starting with this general concept, many parameters of the architecture are subject to an investigation on how they affect the achievable performance and the utilization of the accelerator. Among those design space parameters are the details of the cache hierarchy, concerning sizes and distinction of private and shared caches, the tolerable latency for the direct interconnect between CPU and accelerator, the need to transfer data via this direct link, and the required accelerator size to fit the most beneficial parts of the application on.

Performance estimation and design space exploration for this and other classes of CPU-accelerator architectures are challenging problems. Simulation is the most common approach to evaluate the architectural integration of reconfigurable accelerators before prototyping. While a pure simulation or co-simulation approach provides some insight, its time-consuming design process often limits it to assume a specific interface and a hardware/software partitioning that is hand-tailored to the characteristics of this interface. The challenge for an automated design space exploration is that the specifications of the interface affect what parts of the application can be mapped to the accelerator during hardware/software partitioning. We consider this interdependency between interface and partitioning the reason why the systematic exploration of the design space for the architectural integration has so far not received significant attention in research.
The contribution of this project is a new approach to a fast and fully automated performance estimation of CPU-accelerator architectures. By combining high-level analytical performance modeling, code analysis and profiling, and automated hardware/software partitioning, we can estimate the achievable speedup for arbitrary applications executing on a wide range of CPU-accelerator architectures. The intended use of our method is to quickly identify the most promising areas of the large CPU-accelerator design space for subsequent in-depth analysis and design studies. Consequently, we emphasize modeling flexibility and speed of exploration rather than a high accuracy of the estimation method. The main benefit of our method is that it only needs the application source code or LLVM binary and does not require the user to extract any application-specific performance parameters by hand.

Figure 2 gives an overview of the developed tool-flow. We build upon the LLVM compiler infrastructure. The investigated software is compiled into LLVM assembly language, which is the intermediate code representation on which LLVM's analyses and optimization passes work. We make use of those analyses and profiling features and extend them, e.g., by memory instrumentation and cache simulation, in order to characterize the application. The gathered data is used together with a set of the architecture parameters under investigation to build an estimation model that predicts the total runtime of the application with the given parameters as a sum of four components: the execution time of instructions without memory operations, the memory access time and indirect data exchange through the specified memory hierarchy, the time for transferring control between CPU and accelerator, and the time for exchanging register values between those two components via the direct interface. For this estimation model, we find the optimal partitioning of the application and mapping to CPU and accelerator by formulating and solving an ILP. Alternatively, we provide partitioning heuristics for applications where the optimal



solution requires long computation times. Finally, the tool-flow reports the estimated execution time for the found partitioning. Due to short execution times of about a few seconds for the partitioning and performance estimation phase, our approach allows us to repeat this step for a wide range of parameter sets.

References

[1] Kenter, T.; Plessl, C.; Platzner M. and Kauschke, M.: Estimation and Partitioning for CPU-Accelerator Architectures. Presented at *Intel European Research and Innovation Conference (ERIC)*, October 2011.

[2] Kenter, T.; Plessl, C.; Platzner M. and Kauschke, M.: Performance estimation framework for automated exploration of CPU-accelerator architectures. In *Proc. 19th ACM/SIGDA International Symposium on Field programmable gate arrays (FPGA)*, pages 177–180. ACM, February 2011.

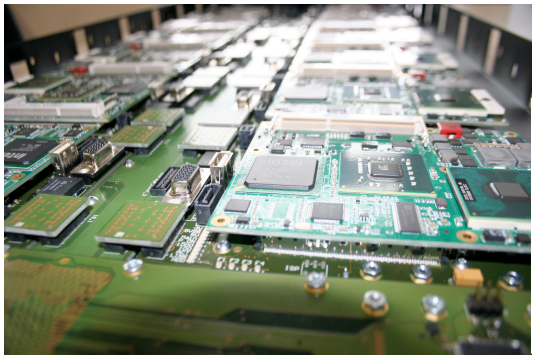
[3] Kenter, T.; Plessl, C.; Platzner M. and Kauschke, M.: Performance estimation for the exploration of CPU-accelerator architectures. In Omar Hammami and Sandra Larrabee, editors, *Proc. Workshop on Architectural Research Prototyping (WARP), Held in conjunction with International Symposium on Computer Architecture (ISCA)*, June 2010.

5.1.4 RECS – Resource Efficient Cluster System

Project coordinator	Prof. Dr. André Brinkmann, Johannes Gutenberg University of Mainz
Project members	Axel Keller, PC². University of Paderborn Matthias Keller, PC². University of Paderborn Dr. Jens Simon, PC². University of Paderborn
Project partner	Christmann informationstechnik + medien GmbH & Co.KG University of Paderborn, Department of System and Circuit Technology
Supported by:	BMWi – Zentrales Innovationsprogramm Mittelstand (ZIM)

General Problem Description

In the project *Resource Efficient Cluster System* (RECS) we aimed to provide supercomputer performance to small and medium-sized enterprises. Typically, these companies do not have large budgets and adequate instructed personal to operate classical supercomputer systems. Our mission was to build affordable and simple administrable computer systems with low operating costs. The systems should scale from small to large configurations to optimally fit application requirements.



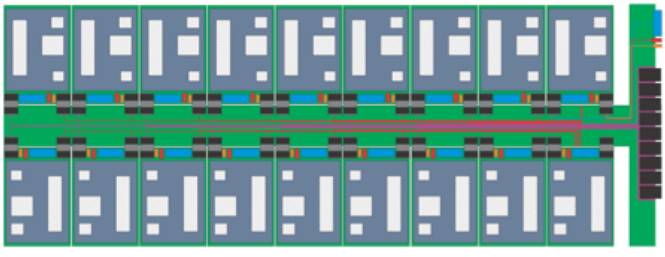


Figure 2: RECS base board architecture and example systems presented at Cebit 2010 and ISC 2010

Problem Details and Work Done

After building a prototype in 2009, we continued the development and were able to present the final system on the Cebit 2010 and the ISC 2010 in Hamburg. The ISC with its trade fair is the most important professional conference and exhibition on High Performance Computing, Networking, and Storage in Europe.

The built system is able to host up to 18 compute nodes with a performance of about 400 GFlop/s in a 1U box. The whole rack may then comprises 600 nodes.

The energy efficiency is about 120 MFlop/s per Watt.

The motherboards were developed by the Department of System and Circuit Technology supporting new cooling concepts. The cluster consists of compute nodes equipped with low power CPUs. The nodes are connected by a standard Gigabit Ethernet network. The system software comprises a Linux operating system with a resource management layer above.

References

[1] ZIM Erfolgsbericht http://www.zim-bmwi.de/zim-koop-foerderbeispiele/zim-koop-025.pdf

[2] ISC2010 http://www.supercomp.de/isc10

5.1.5 System Evaluation, Benchmarking and Operating of Experimental Cluster Systems

Project coordinator	Dr. Jens Simon, PC², University of Paderborn
Project members	Axel Keller, PC², University of Paderborn
	Andreas Krawinkel, PC², University of Paderborn
	Holger Nitsche, PC², University of Paderborn
Supported by	Fujitsu Technology Solutions, ict AG

General Problem Description

In the year 2010, PC² has installed an InfiniBand connected cluster system. The system consists of 60 compute nodes with 720 processor cores and 2160 GByte of main memory. A network Attached Storage (NAS) system is connected with 1 Gigabit-Ethernet to all nodes of the cluster. The capacity of hard-disks of the NAS is 48 TByte. The high-speed interconnect was upgraded from SDR-InfiniBand (10 Gbiut/s) to QDR-InfiniBand (40 GBit/s) in 2011. Emerging technologies, computer systems, interconnects, and software systems have been evaluated by the PC² in the selection phase of the cluster systems and further evaluations are done for the next generation systems. Besides system evaluation and benchmarking of new cluster technologies, different experimental or special purpose cluster systems are operated for research groups of the University of Paderborn.

Problem details and work done

Different computer systems and cluster technologies have been evaluated. The tested systems are up-to date two sockets Intel Xeon systems with quad- and six-core processors, two and four sockets AMD Opteron with hexa-core processors, and some special purpose computer systems with reconfigurable hardware. These systems were equipped with different configurations of high-speed interconnects (InfiniBand quad data rate, 10 Gbit/s Ethernet) and different operating systems of Linux and the Microsoft operating system Windows HPC Server 2008. All benchmarking results are published on the web sides of the Paderborn Benchmarking Center 0.

Co-operations: The PC² benchmarking center is also doing system evaluation and benchmarking for external companies and organizations. The PC² has a long term co-operation with Fujitsu-Siemens Computers where Paderborn acts as a Competence Center for High Performance Computing. System benchmarking is also done for the company ict AG.

References

[1] Simon, J., PC² Benchmarking Center, http://wwwcs.uni-paderborn.de/pc2/about-us/staff/jens-simons-pages/benchmarkingcenter.html

5.1.6 Application Mapping, Monitoring and Optimization for High-Performance Reconfigurable Computing

Project coordinator	Prof. Dr. Marco Platzner, PC², University of Paderborn Dr. Christian Plessl, PC², University of Paderborn
Project members	Tobias Schumacher, PC², University of Paderborn
Supported by:	XtremeData Inc.

General Problem Description

While research has demonstrated the potential of FPGAs as acceleration technology for decades, the creation of accelerators for realistic workloads has been hindered, at least partially, by the lack of commercially available systems. In the last years, however, computing system vendors began to offer machines that combine microprocessors with FPGAs. More recently, FPGA modules that fit into processor sockets have been introduced and provide a fairly standardized way of integrating hardware accelerators into mainstream computing systems.

Developing and optimizing accelerators for such machines is a challenge. Even if FPGA cores for important algorithmic kernels become more and more available, combining them into an overall accelerator remains tricky. Generally, the cores will show data-dependent runtimes and compete for shared resources such as external memory or the host interface, which makes it difficult to decide on a proper number of cores, their topology, the degree of core-level parallelism, data partitioning, etc.

To address these challenges, we have developed IMORC. IMORC is actually two things, an architectural template for creating core-based FPGA accelerators and an on-chip interconnect. The architectural template assists the designer in combining cores to an overall accelerator and greatly facilitates design space exploration, core reuse and portability. The IMORC interconnect relies on a flexible multi- bus structure with slave-side arbitration and offers FIFOs, bitwidth conversion and performance monitoring. Especially performance monitoring is indispensable for debugging and optimizing FPGA accelerators.

Problem Details and Work Done

IMORC assumes applications to be decomposed into a set of communicating cores, which encapsulate computations and access to memory and external communication interfaces.

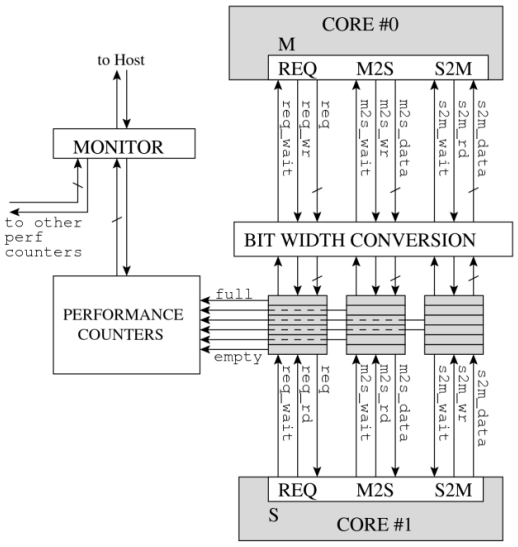


Figure 1: Diagram of an IMORC link

A key element of IMORC is its on-chip interconnect for connecting such cores. Cores in IMORC are connected using links, which are composed of three channels:

1. **REQ** request channel with three fields: one field indicating if the transfer is a read or write, a destination address field and a size field
2. **M2S** data transfers from master to slave
3. **S2M** data transfers from slave to master

Each of the three channels connects master and slave cores using asynchronous FIFOs, enabling both cores to operate in their own clock domains. Additionally, bitwidth conversion modules can be inserted enabling master cores to access slaves at their native data width. Performance counters are inserted, counting the number of times the different FIFOs run full or empty. These values can be monitored in the running system for gathering real performance values with realistic workload.

Additionally to the 1:1 connections presented, IMORC also supports N:1 and N:M connections. For N:1 connections, multiple masters are to be connected to one slave core, which is performed using the IMORC slave arbiters. The slave arbiter monitors the different REQ channels for valid requests and selects an appropriate port, usually in round robin manner. The request is forwarded to the slave core and the corresponding port number together with the request size is forwarded to the read or write datapath, respectively. Here, the appropriate amount of data is transferred from the data channel corresponding to the request.

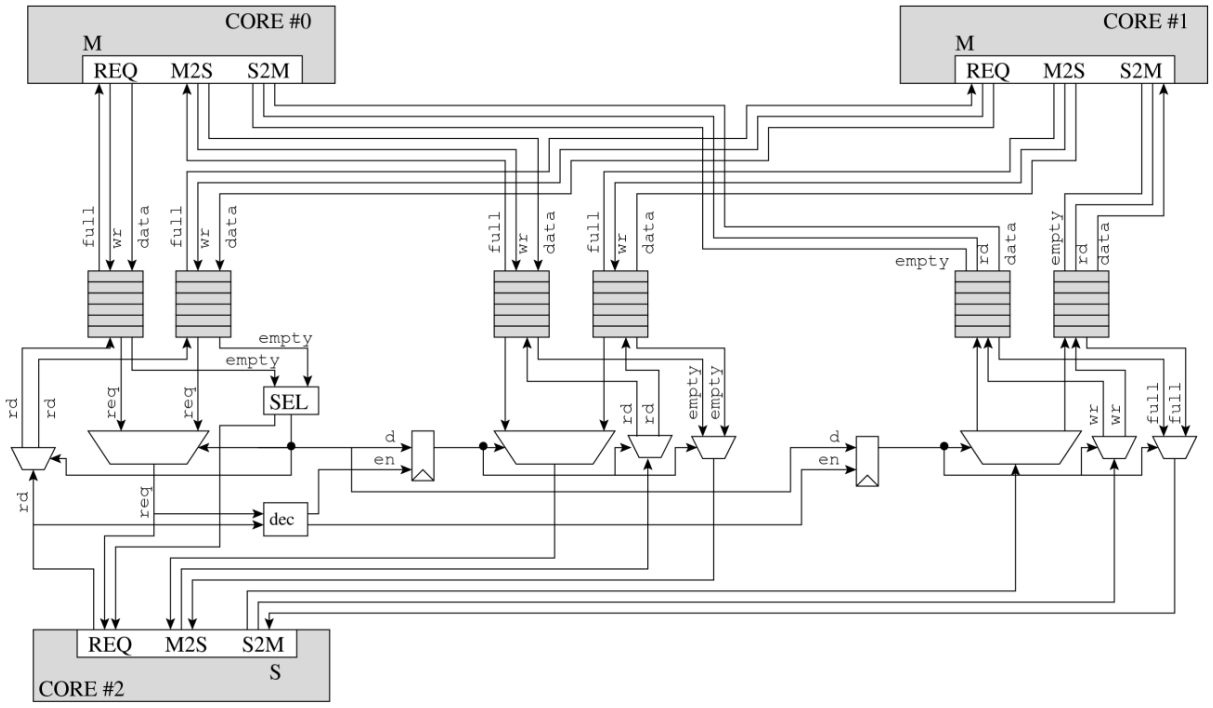


Figure 2: Diagram of an IMORC slave arbiter

1:N connections are supported by using a bus for the FIFOs' wr/rd/wait signals.

This communication scheme provides several benefits to the designer:

- The FIFOs in the links and the separate request and data channels allow designers to decouple the request task from the datapath. The request task can continuously send read/write requests to the slave core, the datapath independently starts operations when data becomes available.
- There is no shared bus forming a central bottleneck.
- High-bandwidth slaves can fulfill the bandwidth requirements of multiple lower-bandwidth master cores, since data is buffered in FIFOs that provide the same bandwidth as the slave's link.
- Cores can transparently access different kinds of memory at their native data width. The bitwidth conversion modules enable cores to access memory of arbitrary width without any information of the concrete memory's layout.

In Addition to the communication infrastructure, IMORC provides a set of utility cores that facilitate the accelerator design:

- **IMORC2REG core:** implements a register block, which is accessible through an IMORC slave interface on the one side and through a native interface on the other side. This interface core can be used for receiving job parameters and presenting them to a core.
- **REGS2IMORC core:** again implements a block of registers, accessible through a native interface. Instead of being accessible through an IMORC slave interface, it

forms an IMORC master and can be used for sending job parameters to other cores.

- **Farming cores:** the IMORC2REGS and REGS2IMORC interface cores form methods for generating jobs to be executed. Often, it is desirable to use multiple instances of a core for executing similar tasks in parallel. The farming cores can take job messages as generated by the REGS2IMORC interface cores and distribute them among multiple IMORC2REGS interface cores for balancing the load between multiple instances of a compute core.
- **Request generator cores:** Many applications need to access data using a predefined or configurable scheme. The request generator cores implement the request path of such cores. The simple form can post a sequence of successive read or write requests with a configurable request size, each. Optionally, using a step parameter the destination address can be incremented further each request, for example for accessing the diagonal elements of a matrix. Additionally, arbitrary sequences of read and write requests can be generated.

These utility cores provide a straight-forward way of generating the request task of accelerator cores, allowing designers to concentrate on the implementation of an optimized datapath. The load sensors which are automatically inserted into the communication infrastructure additionally help in optimizing such accelerators using realistic input data.

Evaluation of the IMORC architectural template was performed on the XtremeData XD1000 reconfigurable platform. The machine provides a dual socket AMD Opteron workstation, where one socket is equipped with an 2.2GHz AMD Opteron processor and the other one with an Altera Stratix II EP2S180 FPGA. Processor and FPGA communicate using a 16bit HyperTransport link running at 800MT/s. 4GB of DDR SDRAM are attached to each the processor and the FPGA. The IMORC support package for this workstation consists of a HT2IMORC interface, which translates HT packets into IMORC packets for CPU initiated communication and additionally allows IMORC to access the CPU's memory using the HyperTransport link. Additionally, an interface to the DDR SDRAM is provided which is based on the Altera DDR SDRAM controller.

In [1], [2] and [3], we give a detailed overview of the IMORC architectural template, our modeling approaches and some case studies demonstrating the potential of this approaches. One of the case studies presented in [2] is an accelerator for the k-th nearest neighbor (KNN) thinning problem.

K-th nearest neighbor methods are widely accepted methods for example in the area of statistics, data analysis and for solving classification problems. The algorithm variant we studied in our work is used for reducing a set of n vectors in a multi-dimensional space to the k vectors that represent the original vectors best. For this purpose, the algorithm starts calculating the distances between all vectors and sorts these distance values in ascending order. The vector with the unique minimum distance to another vector is

discarded and the algorithm starts at the beginning until only k vectors remain.

For the implementation of the IMORC based accelerator, we decomposed the original algorithm into three tasks (distance calculation, sorting, search & discard) and implemented each of these tasks as an IMORC core. The control units of these cores could be implemented nearly completely using the IMORC utility cores, with only a some glue logic and hardly any additional custom control logic. Most of the design time was spent for implementing the datapath.

The design presented in [2] was able to use multiple distance calculator cores and sorter cores, job distribution was performed using the IMORC farming cores. The search & discard core was not parallelized. Figure 3 shows the speedups generated by this accelerator over the original algorithm running on the Opteron CPU of the XD1000 for different configurations (***axb*** means that ***a*** distance calculator and ***b*** sorter cores have been instantiated). The FPGA accelerators were able to achieve a maximum speedup of a factor of 74 over the Opteron host processor.

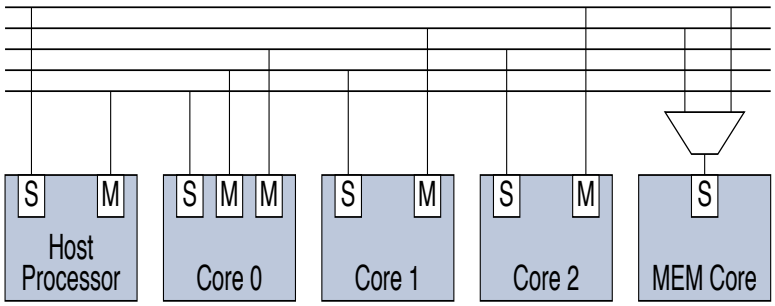


Figure 3: Speedups of the KNN accelerator over the Opteron CPU

Using the IMORC load sensors we also got an insight into the concrete runtime of the different cores. We could see, that for large numbers of vectors the search & discard module took most of the time. This information lately lead to the development of a different search & discard module which is able to be parallelized.

Resource Usage

Development, simulation and synthesis was mainly performed on several fast server systems. For generating different configurations of an accelerator, the Arminius cluster and the Windows HPC cluster was used occasionally. Evaluation of the architectural template was performed on the XtremeData XD1000 reconfigurable workstation.

References

[1] Schumacher, T.; Plessl, C. and Platzner, M.: “IMORC: Application Mapping, Monitoring and Optimization for High-Performance Reconfigurable Computing,” in Proc. IEEE Symp. on Field-Programmable Custom Computing Machines (FCCM ’09). IEEE, 2009.

[2] Schumacher, T.; Plessl, C. and Platzner, M.: “An Accelerator for k-th Nearest Neighbor Thinning based on the IMORC Infrastructure”, in *Proceedings of the 19th International Conference on Field Programmable Logic and Applications (FPL)*, Prague, Czech Republic, August/September 2009. IEEE

[3] Schumacher, T.; Süß, T.; Plessl, C. and Platzner, M.: “Communication Performance Characterization for Reconfigurable Accelerator Design on the XD1000 ”, in Proc. Int. Conf. on ReConFigurable Computing and FPGAs (RECONFIG '09)

5.2 Grid Technologies

5.2.1 HPC Cloud

Project coordinator	Prof. Dr. André Brinkmann, PC², University of Paderborn
Project members	Matthias Keller, PC², University of Paderborn

General Problem Description

The PC^2 is a service provider for high performance computing infrastructure. Despite the easy access, users need some degree of technical knowledge about the underlying software-system to use the provided infrastructure. Such a degree of technical understanding has to be learned previously. Especially for (external) non-computer-scientists this is a first serious hurdle on the way to access the provided computing power. Minimizing that hurdle and simplifying the first steps and the whole workflow will increase the quality of the provided service. When the need for technical knowledge is eliminated, the number of potential users, particularly non-computer-scientists, will increase. Within this project, a prototype web service has been developed in cooperation with external non-computer-scientists for an iterative refinement of the software to thoroughly comprehend the requirements, to solve emerging issues, and finally to elevate the overall quality of the service.

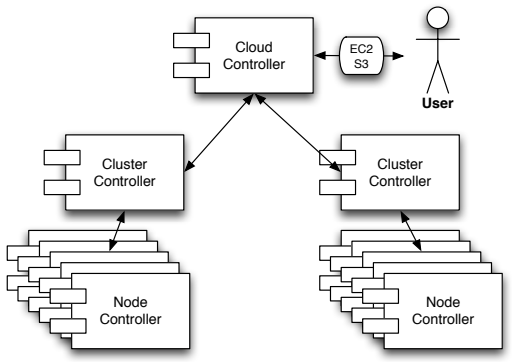


Figure 4: shows the 3-tiered architecture of eucalyptus.

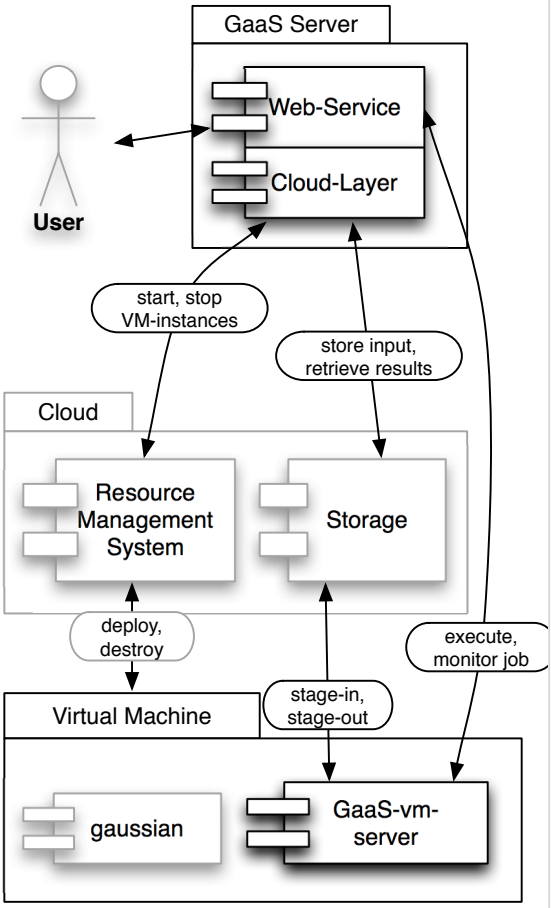


Figure 5: depicts the software architecture of Gaussian as a service. Components in bold are part of the prototype.

Beside well known Middleware Systems for Grid Computing, like UNICORE, Globus Toolkit, or gLite, another Middleware System, a commercially driven alternative approach for utilizing compute resources, is chosen relating to Cloud Computing [1]. While Cloud computing currently gains more and more importance and influence in the industry, it lacks standardizations and in depth academic exploration and research. Except for the similarity of utilizing hardware resources with Grid computing, Cloud computing has four major differences in the context of this project: Service, Access, Virtualization, and Scale. The first two differences are emphasized by the service-oriented perspective, which insinuates a high-level abstraction or interface (generally a http-request), which provides an easy and ready-to-use access to the provided services of the cloud. Those services are often called X-as-a-Service, like Infrastructure-, Platform-, or Software-as-a-Service, which are not invented by Cloud-computing but mostly used in its context. Due to the unique use case of high performance computing, a High-Performance-Computing-as-a-Service as proposed by the Steinbuch Centre for Computing [2] is imaginable. Privileges to access the service are usually acquired by creating a web account, which is, therefore, substantially easier than applying for Grid Credentials through a personalized administrative process. Another

inherent concept of Cloud computing is virtualization. In contrast to the Grid Computing Community, which made some efforts to only extend a grid with virtualization capability, Cloud Computing takes the opposite approach; only virtual machines exist and applications or computations run within them. Another difference is the vision of the enormous scale of a cloud, deducing on the one hand the demand for a very scalable architecture and on the other hand a cost efficient management of those resources ,for example, by choosing a low maintenance cost place or by automating the administration.

Problem details and work done

The main goal is to lower the technical barrier in respect to performing a job-workflow and receiving privileges. Particularly for the first aspect, the developed software needs to be specialized for this type of job, for the used application, or for a set of similar applications.

The major contribution was the development of a flexible architecture and prototype [5], which enables easy adjustments on different applications and extensions for future work. Therewith, a user can access a web-form to upload the input data for the job and later retrieve the results.

In the back-end, a cloud software solution manages a dedicated cluster. Different software solutions (namely Eucalyptus [4], OpenStack, and OpenNebula) were installed to gain first-hand experiences in these new technologies.

Resource Usage

For a realistic test environment for test users to launch compute jobs, the “Paderborn HPC Cloud” cluster was utilized.

References

[1] Mc Evoy, G.V. and Schulze, B.: "Using clouds to address grid limitations," in *MGC '08: Proceedings of the 6th international workshop on Middleware for grid computing*. New York, NY, USA: ACM, 2008, pp. 1-6. [Online]. Available: http://dx.doi.org/10.1145/1462704.1462715

[2] Steinbuch Centre for Computing. Karlsruher Institute of Technology. About High Performance Computing as a Service. [Online]. Available: http://www.scc.kit.edu/forschung/4942.php

[3] Gaussian Website, http://www.gaussian.com/g_prod/g09.htm
[4] Nurmi, D.; Wolski, R.; Grzegorczyk, C.; Obertelli, G.; Soman, S., Youseff, L. and Zagorodnov, D.: "The eucalyptus open-source cloud-computing system," in *Proceedings of 9th IEEE International Symposium on Cluster Computing and the Grid, 2009*. [Online]. Available: http://open.eucalyptus.com/documents/ccgrid2009.pdf
[5] Keller, M., Meister, D., Brinkmann, A., Terboven, C., & Bischof, C. (2011). eScience Cloud Infrastructure. 2011 37th EUROMICRO Conference on Software Engineering and Advanced Applications (pp. 188-195). Ieee. doi:10.1109/SEAA.2011.38

5.2.2 MoSGrid – Molecular Simulation Grid

Project coordinator	Ulrich Lang, University of Cologne
Project members	Lars Packschies, Dirk Blunk, Sebastian Breuers, University of Cologne Oliver Kohlbacher, Sandra Gesing, Eberhard-Karls-Universität Tübingen Jun.-Prof. Dr. André Brinkmann, PC^2, University of Paderborn Georg Birkenheuer, PC^2, University of Paderborn Prof. Dr. Gregor Fels, Department Chemie, University of Paderborn Dr. Jens Krüger, Department Chemie, University of Paderborn Dr. Sonja Herres-Pawlis, Department Chemie, Technische Universität Dortmund Dr. Alexander Reinefeld, Patrick Schäfers, Konrad-Zuse-Zentrum für Informationstechnik, Berlin Ralph Müller-Pfefferkorn, Richard Grunzke, Technische Universität Dresden Bayer Technology Services GmbH, Leverkusen Origines GmbH, Martinsried GETLIG & TAR, Falkensee BioSolveIT, Sankt Augustin COSMOlogic GmbH & Co.KG, Leverkusen
Supported by:	BMBF, Bundesministerium für bildung und Forschung – Project grant 01IG09006

General Problem Description

The chemical industry is one of the most research-intensive sectors of the German economy. The high level of innovative dynamism fosters close cooperation between industry and scientific institutions. The MoSGrid (Molecular Simulation Grid) should generate competitive advantages for this sector of industry and science through the Grid. In MoSGrid, the key focus is on setting up and providing grid services for performing molecular simulations. MoSGrid makes the D-Grid infrastructure available for high-performance computing in the area of molecular simulations, including the annotation of metadata results and the provision of methods for data mining and knowledge generation. The aim of MoSGrid is to support the user in all stages of the simulation process. A portal provides access to data repositories that store information about calculated molecular

properties as well as ‘recipes’ – standard methods for the provided applications. With the aid of these recipes, application-specific input files and computing requests are automatically generated and subsequently submitted to the Grid (pre-processing and job submission). Furthermore, users are supported by an evaluation of the calculation results. This facilitates the preparation and processing of data for further calculations and analyses that derive from it. Additional knowledge is attained by cross-referencing different results’ data files. Furthermore, the data repository allows external referencing of simulation results.

The D-Grid initiative already enables the supported communities to gain simple access to shared computing resources. Based on this technology and the tools, MoSGrid integrates the special requirements of chemically oriented scientists into the D-Grid infrastructure. The high complexity of this discipline’s software (e.g., quantum mechanics or molecular dynamics) often makes using this technology difficult for non-specialist scientists. This difficulty is compounded by the fact that user interfaces such as graphic accessibility function, are often not available or are inadequate. A clear method selection and simple importing of molecular data, as well as the automatic set-up of a program-specific input data, assist the user. Consequently, MoSGrid offers a web-based, graphic user interface, which enables the transparent use of the installed applications. Therefore, high-quality standard techniques are suggested on request, e.g., for basic structure optimisation with quantum chemical methods or standard workflows for molecular dynamic research, which scientists can modify based on their own requirements. From the information received, the input data is generated automatically for the actual simulation, supported by adapters. Based on well-known and established methods, jobs are submitted to the Grid and supervised. The adapters are created, maintained, and expanded by the consortium and the users. Simulation results are automatically extracted after the calculations are completed, assisted by a suitable parser adapted to the special output formats of the different programs, and checked for elementary plausibility (post-processing). At the user’s request, these results are transferred to collaborative data repositories of molecular properties.

Simple access to shared data is, along with the common use of computing capacity, a fundamental basis for the acceptance of grids in business and science. MoSGrid sets up the technological basis in order to provide results of extensive molecular simulations that can subsequently be used for data mining processes. Parsers aid the generation of the result data sets. In addition, data repositories are developed and operated. They support scientists through coordinated access to simulation data and the information derived from it to find solutions to complex questions. As a consequence, the generation of metadata is an important goal for MoSGrid in applying simulation results to complex searches and logical operations. For this, well-known ontologies are used and extended for specific requirements through MoSGrid.

The planned data repositories are of practical importance because they allow users within and outside of MoSGrid to access the expertise of those already applying simulations. Topic-specific data is derived from molecular simulations for the identification of relationships between structural properties. MoSgGid supports:

- Fundamental research - investigating experimental reaction phenomena
- Applied research - optimisation of materials
- Product-related development - classification of potentially bioactive agents.

These broadrange of topics are also shown through the participation of notable business partners in MoSGrid.

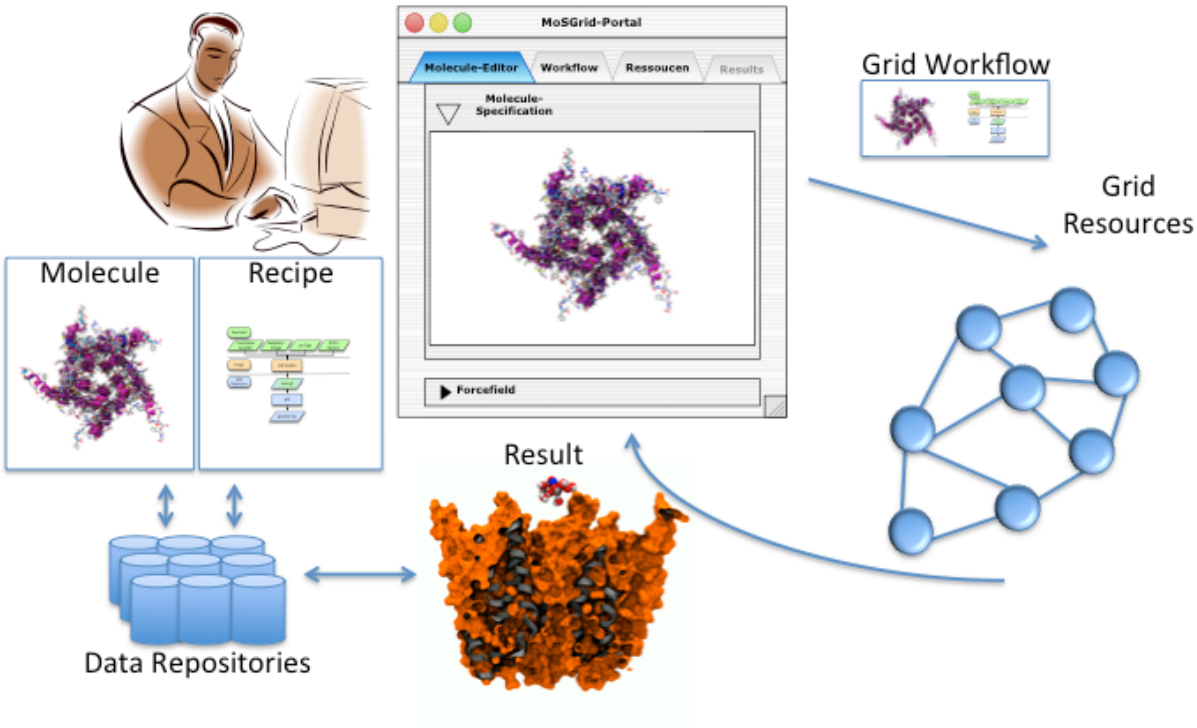


Figure 1: MoSGrid in a nutshell

The value of the MoSGrid project for business and science relies on the quality, attractive content, and sufficiently broad coverage of data, which are only financially possible through the high throughput of computing scenarios in the Grid. The breadth of expert knowledge is available to the MoSGrid thanks to the participating partners from both business and scientific communities.

MoSGrid started in September 2009 and has a duration of 36 month. From the Universität Paderborn, the PC^2 as well as the chemistry WG Fels and WG Pawlis were part of the project.

Problem Details and Work Done

The cooperation between the PC^2 and the Chemistry Department of the Universität Paderborn in scope of Grid computing was based on a cooperation on a UNICORE6 adaption of Gaussian. Later, the cooperation resulted in the MoSGrid project.

Work Package 1

The project- and community management was lead by the Chemistry Department. The PC^2 supported the chemists in computer science questions and tasks. The main tasks were the analysis of user requirements, communication with beta testers, work on the newsletters, and the organisation of the community meetings.

Work Package 2

In the beginning of 2010, we started with the requirements' analysis for the portal technology. Here, the PC^2 supported the Universität Tübingen in the review of portal solutions. LiferyLiferay, http://www.liferay.com/web/guest/partners/sun. was, in cooperation with WS-PGRADEWS-PGRADE, https://guse.sztaki.hu/liferay-portal-6.0.5/, chosen as the best solution.

Thereafter, the PC^2 was the leader in the development process of the user portlet for molecular dynamics calculations. This was done in strong collaboration with the Chemistry Department of the Universität Paderborn. The chemists supported the development process by specifying requirements and testing the beta software.

In 2010, the generation one of the MD-portlet used the uccAPI interface to connect to the UNICORE workflow infrastructure. In 2011, the connection was changed to the gUSE workflow engine through the application specific module (ASM) interface. Also in 2011, the knowledge of the development of the MD portlet was transferred to the development of the docking portlet concept.

Work Package 4

This work package contains two main tasks. The first one was the development of workflows that should cover the simulation of chemical recipes. The second task was the selection of suitable workflow tools and a creation of interfaces that allow an easy connection of the portlets to the selected workflow engines.

Concepts for Workflows

The aim of the PC^2 for the workflow creation was to provide a collection of recipes to the user of the MoSGrid portal. This was done in close collaboration with the participating chemists. They are familiar with recipes from their daily work. In this context, a recipe is a concept that should be implemented using a workflow. The workflows can be depicted as a multi-step process comprising of the following tasks:

- A job definition task describes the chemical task the user wants to solve.
- A metaprocessing task checks the user input for consistency

- A preprocessing task translates the chemical Meta information to an application specific input format.
- The job submission task can include several serial and/or parallel simulations.
- The steering of the job relies on a monitoring and job abortion mechanism.
- A postprocessing task extracts application independent information from the result files.

The huge benefit of these workflow steps is that the process of designing a molecular simulation is separated from the application. The chemists select the kind of discipline (e.g., quantum chemistry, molecular dynamics, or docking) and apply their recipe. Then, the MoSGrid software selects the appropriate simulation codecs. Thus, the user can concentrate on the definition of the scientific part of the simulation while MoSGrid cares about the tedious steps of preprocessing, job submission, and postprocessing in a totally transparent way.
In addition to the general workflow concept, several exemplary workflows were developed by UPB.

Selection of the Software
For the selection of a suitable workflow engine, different workflow systems were compared to the requirements for the simulation of chemical recipes. As a result of the evaluation, the UNICORE 6 integrated workflow-engine and the engine of WS-PGRADEWS-PGRADE, https://guse.sztaki.hu/liferay-portal-6.0.5/ were usable.
It was decided to first integrate the UNICORE engine in the project because it was easy to connect. Secondly, the WS-PGRADE workflow engine of gUSE was adapted because it already supported the WS-PGRADE portal technology and offered an integrated workflow editor. The underlying grid User Support Environment (gUSE) contained a data-driven workflow engine that expresses the dependencies of single steps in a workflow by directed acyclic graphs (DAGs). The workflow engine encapsulates the single steps and invokes submitters (Java based applications) for each job. Via these submitters, gUSE offers the possibility to submit jobs to grid middleware environments like Globus Toolkit and gLite, desktop grids, clouds, clusters, and unique web-services.

Development
In 2010, the PC^2 developed an interface library to allow the MoSGrid portlets a direct access to the UNICORE 6 workflow engine. For the interface, the abilities of the UNICORE 6 command-line client UCC and the UNICORE High Level API were compared. Due to the workflow abilities, the UNICORE 6 command-line client UCC was chosen. The resulting interface encapsulated the abilities of the UCC and, therefore, was named uccAPI. The usability of the uccAPI was demonstrated by the generation 1 of the MoSGrid MD-portlet. The portlet submitted workflows directly to the UNICORE workflow engine.

Work Package 5

The PC^2 offered the BISGrid cluster system for the integration of the simulation codes for MosGrid. The supported codecs were Gaussian, NWChem, and Gromacs. Additionally, the PC^2 supported other partners in UNICORE issues.

In 2010, Sonja Herres-Pawlis moved from Paderborn to the TU Dortmund and in 2011, André Brinkmann moved to the University of Mainz, Gregor Fels retired, and Jens Krüger moved to University of Tübingen. The University of Paderborn remains as an unfounded full partner in the project.

Project results were published in [3 - 9].

References

[1] Liferay, http://www.liferay.com/web/guest/partners/sun.

[2] WS-PGRADE, https://guse.sztaki.hu/liferay-portal-6.0.5/

[3] Grid-Workflows in Molecular Science. Georg Birkenheuer and Sebastian Breuers and André Brinkmann and Dirk Blunk and Gregor Fels and Sandra Gesing and Sonja Herres-Pawlis and Oliver Kohlbacher and Jens Krüger and Lars Packschies. Proceedings of the Grid Workflow Workshop (GWW), March 2010

[4] MoSGrid: Progress of Workflow driven Chemical Simulations. Birkenheuer, G., Blunk, D., Breuers, S., Brinkmann, A., Fels, G., Gesing, S., Grunzke, R., Herres-Pawlis, S., Kohlbacher, O., Krüger, J., Lang, U., Packschies, L., Müller-Pfefferkorn, R., Schäfer, P., Schuster, J., Steinke, T., Warzecha, K., and Wewior, M. GWW 2011 (Grid Workflow Workshop), Cologne, Germany, March 2011.

[5] A Science Gateway for Molecular Simulations. Gesing, S., Kacsuk, P., Kozlovszky, M., Birkenheuer, G., Blunk, D., Breuers, S., Brinkmann, A., Fels, G., Grunzke, R., Herres-Pawlis, S., Krüger, J., Packschies, L., Müller-Pfefferkorn, R., Schäfer, P., Steinke, T., Szikszay Fabri, A., Warzecha, K., Wewior, M., and Kohlbacher, O. In: EGI User Forum 2011, Book of Abstracts, pp. 94–95, ISBN 978 90 816927 1 7, April 2011.

[6] Granular Security for a Science Gateway in Structural Bioinformatics. Gesing, S., Grunzke, R., Balasko, A., Birkenheuer, G., Blunk, D., Breuers, S., Brinkmann, A., Fels, G., Herres-Pawlis, S., Kacsuk, P., Kozlovszky, M., Krüger, J., Packschies, L., Schäfer, P., Schuller, B., Schuster, J., Steinke, T., Szikszay Fabri, A., Wewior, M., Müller-Pfefferkorn, R., and Kohlbacher, O. IWSG-Life 2011 (International Workshop on Science Gateways for Life Sciences), London, UK, June 2011.

[7] Molecular Simulation Grid. Jens Krüger, Georg Birkenheuer, Dirk Breuers Sebastian Blunk, André Brinkmann, Gregor Fels, Sandra Gesing, Richard Grunzke, Oliver Kohlbacher, Nico Kruber, Ulrich Lang, Lars Packschies, Ralph Herres-Pawlis Sonja Müller-Pfefferkorn, Patrick Schäfer, Hans-Günther Schmalz, Thomas Steinke, Klaus-

Dieter Warzecha, and Martin Wewior. German Conference on Chemoinformatics, Goslar, 2010

[8] Workflow Interoperability in a Grid Portal for Molecular Simulations. Sandra Gesing, Istvan Marton, Georg Birkenheuer, Bernd Schuller, Richard Grunzke, Jens Krüger, Sebastian Breuers, Dirk Blunk, Georg Fels, Lars Packschies, Andre Brinkmann, Oliver Kohlbacher, and Miklos Kozlovszky. In Roberto Barbera, Giuseppe Andronico, and Giuseppe La Rocca, editors, Proceedings of the International Workshop on Science Gateways (IWSG10), pages 44–48. Consorzio COMETA, 2010.

[9] The MoSGrid Gaussian Portlet – Technologies for the Implementation of Portlets for Molecular Simulations. Martin Wewior, Lars Packschies, Dirk Blunk, Daniel Wickeroth, Klaus-Dieter Warzecha, Sonja Herres-Pawlis, Sandra Gesing, Sebastian Breuers, Jens Krüger, Georg Birkenheuer, and Ulrich Lang. In Proceedings of the International Workshop on Science Gateways (IWSG10), pages 39–43. Consorzio COMETA, 2010.

5.2.3 DGSI – D-GRID Scheduler Interoperability

Project coordinator	Bernhard Schott, Platform Computing GmbH, Ratingen
Project members	This Metsch, Platform Computing GmbH, Ratingen Martin Hofmann-Apitius, Bonn-Aachen International Center for Information Technology (BIT) Wolfgang Ziegler, Fraunhofer-Gesellschaft zur Förderung der angewandten Forschung e.V. Oliver Wäldrich, Fraunhofer-Gesellschaft zur Förderung der angewandten Forschung e.V. Andreas Hoheisel, Fraunhofer-Gesellschaft zur Förderung der angewandten Forschung e.V. Oleg Khovalko, Fraunhofer-Gesellschaft zur Förderung der angewandten Forschung e.V. Ulrich Schwardmann, Tim Ehlers. Dietmar Sommerfeld, GWDG, Göttingen Jun.-Prof. Dr. André Brinkmann, PC^2, University of Paderborn Georg Birkenheuer, PC^2, University of Paderborn Axel Keller, PC^2, University of Paderborn Uwe Schwiegelshohn, Andreas Fölling, Alexander Papaspyrou, Technische Universität Dortmund Rainer Spurzem, Klaus Rieger Ruprecht-Karls-Universität Heidelberg Helmut Heller, Emmanouil Paisios, Bayrische Akademie der Wissenschaften, Leibniz-Rechenzentrum Wolfgang E. Nagel, Technische Universität Dresden
Supported by:	BMBF – Bundesministerium für Bildung und Forschung – Project grant 01IG09009

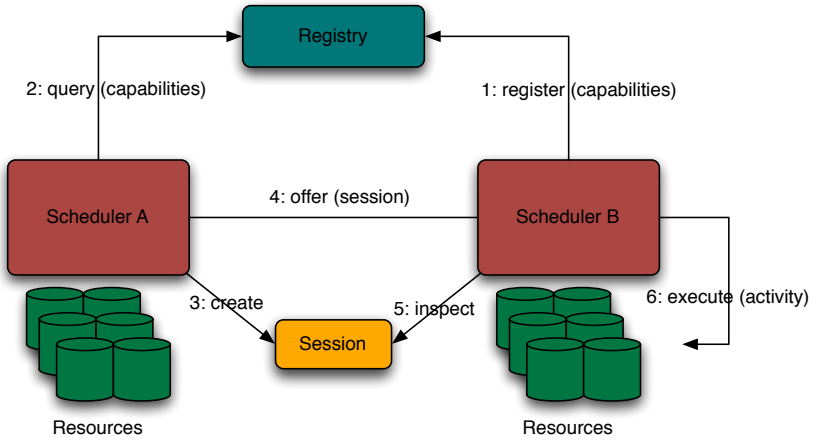
General Problem Description

Most Service Grids have the ability to efficiently distribute user workload to the available resources. This issue, usually generalized under the term *Grid Scheduling* or *Meta Scheduling*, is already very diverse within a community: both submitted jobs and available resources differ considerably, to the extent that their coordination requires specialized knowledge about usage scenarios and infrastructure. This leads to very different,

community-specific approaches for the development of Grid scheduling services. The resulting incompatibility on the meta-scheduling level, however, proved to be a major hurdle for the coordinated cooperation of different Service Grids especially when focusing on the overall goal of better resource utilization. Moreover, cooperations on a scientific, cross-disciplinary level are being impaired as well. As such, two major use cases for temporarily including alien resources into the own community arise: the need to cover peak demand and the usage of specialized resources.

The **D-Grid Scheduler Interoperability (DGSI)** targets these use cases with the conception and development of an interoperability layer for Grid level scheduling in service Grids. DGSI allows the scheduler of a community to distribute the workload among resources within the management domain of another community while the individual, specialized scheduling solutions are still run by the communities. DGSI offers new perspectives for community collaboration, resource sharing, and efficient utilization. For sharing the resources between different communities, we follow two different approaches: activity delegation and resource delegation.

In activity delegation, a Grid scheduler hands over an activity and the management of its execution to the domain of another community's scheduler. This approach also requires negotiation between both Grid scheduling services.



Resource delegation has become an interesting alternative to activity delegation due to its advantages, like support for local requirements, e.g., special workflow scheduling tools or better control over the delegated resources. At the same time, business models for resource delegation have been evolving, and providing Cloud resources to customers is based on similar delegation mechanisms.

The management of delegated resources is exclusively realized on top of the base middleware. The result of a negotiation between two community schedulers is a virtual middleware interface to the resources with the following tasks:

- Encapsulate the delegated resources
- Monitor the agreed terms of usage
- Hide the security-relevant adaptations and the delegation of rights

In the negotiation phase, following the initial resource discovery phase, the community scheduler initiates the delegation negotiations with other community schedulers. They represent the access points to the detected resources. For the negotiation and the agreement on the terms of the resource delegation, WS-Agreement [1] has been selected as the protocol and language for creating Service Level Agreements. WS-Agreement is a specification of the Open Grid Forum (OGF), developed and maintained by the Grid Resource Allocation Protocol Working Group (GRAAP-WG) [2].

DGSI started in June 2009 and has a duration of 36 months. The ideas of the project were presented at “Infrastructure Federation Through Virtualized Delegation of Resources and Services. Georg Birkenheuer, André Brinkmann, Mikael Högqvist, Alexander Papaspyrou, Bernhard Schott, Dietmar Sommerfeld, and Wolfgang Ziegler. In the Journal of Grid Computing, 2011.

Problem Details and Work Done

The main work of the PC^2 was done for the work package 3 „Delegation von Ressourcen“. The PC^2 was the work package leader in this WP.

Resource Delegation Software Generation 1

The main work of the PC^2 in DGSI in 2010 was to create and implement the resource delegation RD software. In generation 1, the task was the setup of the proxy and the virtualization approach. This work was realized through the concept of the Delegation Daemon and the Virtual Front End.

The duties of the PC^2, firstly, contained the interface and the software architecture specification for RD and, secondly, the implementation of the key components. The components were the WS-Agreement interface, the Delegation Daemon scheduling component (DD), the Advanced Reservation Service (ARS), and the Virtual Front End (VFE).

The component communicating with the Meta schedulers is the interface. It implements the WS-Agreement specification[1]. The WS-Agreement was chosen as the interface because it had already been used by the activity delegation and was also supported by SLA4D-Grid. This allowed reusing concepts and software from the activity delegation. The interface was embedded in an Apache web service to be accessible by the Meta schedulers.

The managing component for a RD is the delegation daemon-scheduling component. It is responsible for the management of the VFEs. It reserves resources with the Advanced Reservation Service ARS and plans the VFE setup. The DD is also responsible for starting, stopping, and cleaning up the VFEs.

The component for creating the advanced reservations is the ARS. It is responsible for creating and managing reservations on the RMS. For generation 1, the ARS supported Torque and Maui.

The VFE machines are the access points to a resource delegation by the foreign Meta scheduler. The VFEs were created by the VFE setup component, which was created by the GWDG. The PC^2 supported the colleagues in the specification of the requirements of the virtual grid middleware. In addition, there was a close cooperation in testing the virtualization environment, customizing the software, and in troubleshooting.

All components, except the VFE setup, communicated through an XML-RPC Web Service. The VFE setup was a shell-based component that was started by the DD.
The software of the RD generation 1 was published in milestone M3-4 and in JournalOfGridComputing [3].

Resource Delegation Software Generation 2

In 2011, the software of the RD approach in generation 2 was adapted to use the SLA negotiation framework from SLA4D-Grid. SLA4D-Grid components were responsible for the steps preceding a negotiation and the creation of the advanced reservations on the RMS.
An additional component, the steering job, was introduced to manage the Virtual Front Ends.
The steering job included the tasks formally managed by the DD. The steering job starts the VFE setup with the information about the RD reservation from SLA4D-Grid and

[1] WS-Agreement specification: http://www.ogf.org/documents/GFD.107.pdf

manages the VFE during the delegations. The colleagues of the GWDG were supported in the extension of the VFE creation process to new requirements.

Work for other work packages

Besides the work for the RD, the PC^2 supported the other work packages with knowledge and feedback. Another important task of the PC^2 in WP-1 was the main task T4 “Interoperabilität der Einhaltung von Leistungszusicherungen”.

The PC^2 was the task leader. Aim of T4 was to define Key Performance Indicators, KPIs, for the application for the activity and resource delegation. (KPI)s are an instrument to monitor and check if performance guarantees are held. The evaluated measures were documented in the deliverable D1-6 “Specification of Interoperable Performance-Confirmations for Delegations“ and presented at the OGF.

References

[1] WS-Agreement specification: http://www.ogf.org/documents/GFD.107.pdf

[2] GRAAP-WG: http://www.ogf.org/gf/group_info/view.php?group=graap-wg

[3] Infrastructure Federation Through Virtualized Delegation of Resources and Services. Georg Birkenheuer, André Brinkmann, Mikael Högqvist, Alexander Papaspyrou, Bernhard Schott, Dietmar Sommerfeld, and Wolfgang Ziegler. In the Journal of Grid Computing, 2011.

5.2.4 EDGI Project

Project coordinator	Prof. Dr. Peter Kascuk, Laboratory of parallel and distributed Systems (LPDS), Hungary
Project members	Jun.-Prof. Dr. André Brinkmann, PC², University of Paderborn Matthias Keller, PC², University of Paderborn MTA SZTAKI (HU), INRAI (F), CNRS (F) UoW (UK), CU (UK), Unizar-Ibercivis (ES), FCTUC (PT), AlmereGrid (NL), UPB (DE), UCPH (DK)
Supported by	Funded by the FP7/2007- 2013 under grant agreement no 261556.

General Problem Description

The Project European Desktop Grid Initiative (EDGI) develops middleware to extent Service Grids, like ARC, gLite, or UNICORE, with Desktop Grids (DGs) in order to support European Grid Initiative (EGI) and National Grid Initiative user communities. These are heavy users of Distributed Computing Infrastructures (DCIs) and require an extremely large number of CPUs and cores. EDGI will go beyond existing DCIs, which are typically Cluster Grids and Supercomputer Grids, and will extend them with public and institutional Desktop Grids and Clouds. EDGI will integrate software components of ARC, gLite, UNICORE, BOINC, XWHEP, 3G Bridge, and Cloud middleware, such as OpenNebula and Eucalyptus, into SG-DG-Cloud platforms for service provision, and as a result EDGI will extend ARC, gLite, and UNICORE Grids with volunteer and institutional DG systems.

EDGI [1] started in June 2010 and was extended until April 2012. The International Desktop Grid Federation (IDGF) [2] will perform further DG activities.

Problem details and work done

The PC^2 focuses on the UNICORE side: extending the UNICORE system so Desktop Grids can be accessed as an abstract compute resource and providing support for UNICORE communities to pave the way towards Desktop Grids via UNICORE.

The EDGI bridging mechanism (and Desktop Grid environment) has been integrated into the UNICORE server to forward jobs to Desktop Grid servers [3]. The UNICORE extension

will be part of future EMI UNICORE releases to ease deployment and operation for other compute centers.

The PC^2 further provides consulting for interested UNICORE communities on how to access these additional and new kinds of compute resources. In a joint effort, Desktop Grid resources via UNICORE can be provided as normal D-Grid resources despite their different technical nature. This enables a common and easy way for German researchers to access these new resources.

Resource Usage

For development and provisioning of productive systems, computer resources of the PC^2 are used.

References

[1] EDGI Project Website: http://edgi-project.eu/introduction
Details: http://edgi-project.eu/downloads/-document_library_display/7FkI/view/25605
Partners: http://edgi-project.eu/partners

[2] IDGF Website: http://desktopgridfederation.org/

[3] Keller, M.; Kovacs, J.: A. B. (2011). Desktop Grids Opening up to UNICORE. In M. Romberg, P. Bala, R. Müller-Pfefferkorn, & D. Mallmann (Eds.), *UNICORE Summit 2011 Proceedings* (pp. 67-76). Torun, Polan: Forschungszentrum Jülich GmbH Zentralbibliothek, Verlag. Retrieved from http://juwel.fz-juelich.de: 8080/dspace/handle/2128/4518

5.2.5 HYDRA – Network embedded system middleware for heterogeneous physical devices in a distributed architecture

Project coordinator	Prof. Dr. André Brinkmann, PC², University of Paderborn
Project members	Dr. SaschaEffert, PC², University of Paderborn Yan Gao, PC², University of Paderborn Dirk Meister, PC², University of Paderborn
Supported by	European Commission (IST – 2005-034891)

General Problem Description

Information technologies are nowadays spread in almost every area of modern life. Not only the desktop pc or notebook is part of this movement, but also more and more embedded systems, integrated in house automation, cars or even clothes. To improve the value of such embedded devices Hydra aims to build up on these devices an “Internet of Things”.

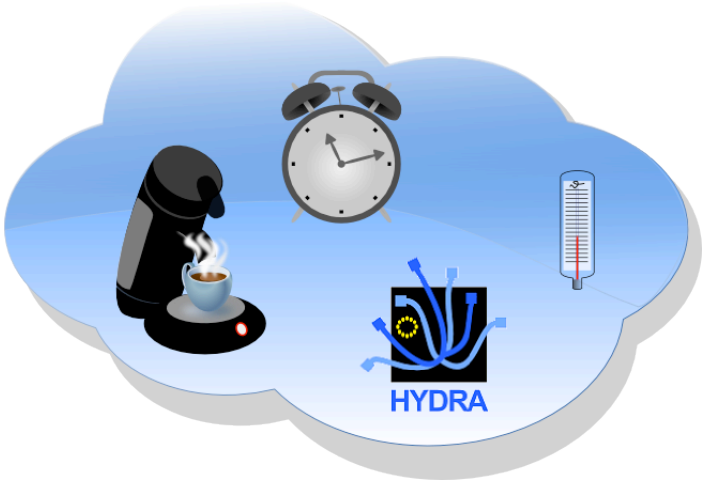


Figure 1: Hydra connecting home devices

“Internet of Things” means, that devices are reachable in an easy way over a network by remote applications. Manufactures designing a hydra enabled device have to care about adding the device to the network. For small devices, like sensors, this can be realized using a Hydra proxy. Gateways are responsible to build a connected network even over different physical network types like Ethernet, Bluetooth or Zigbee.

The Hydra project aims to build a middleware to support this kind of Peer-to-Peer network. It delivers technologies to build virtual devices upon physical hardware description, to

develop applications being able to find devices and to communicate with them in a network independent and secure way.

The example described in Figure 1 shows how the Hydra middleware can improve the value of common devices. In most modern homes are a couple of independent devices, e.g. a coffee maker, an alarm clock and temperature sensors for the heater. Connecting these devices the alarm clock would be able to ring earlier in the winter when the streets are frozen by connecting the temperature sensor. More over it could tell the coffee maker to brew up fresh coffee. But Hydra is not only focused on the home sector but also on the other ones like healthcare and agriculture.

The PC^2 is working in the area of storage in this highly dynamic P2P network. The demands in storage are very special in this area. Devices join and leave the network dynamically, often even without disconnecting. Nevertheless data should be reachable in the network. Sometimes it is also important to keep data locally on devices available. This means applications need to be location aware in the place data is stored. Therefore, if an application needs more storage then the local device has, it can access storage on other devices, e.g. a mobile can store downloaded pictures directly on a camera.

Data should also be stored in a robust way, so that no data loss will happen because some devices fail or are no more reachable. Therefore sophisticated replication and synchronization algorithms are needed, which are able to discover and repair failures fast.

The area of security is also very important in Hydra. Data sent over the network has to be protected, so that other members of the network cannot change or view the data. But it is also necessary to store the data on the devices in a secure way, so that only privileged users are able to read it. This way a user can use not trusted devices to store even sensible data without having to bother the owner of the device could access his data.

Problem details and work done

As shown in Figure 2 a number of devices were developed to realize the key functionalities of the Hydra middleware. One of the most important managers is the Network Manager, which is responsible for building a network aware way for communication between devices and applications. Therefore the Network Manager builds virtual representations of managers or devices (entities) on a local device. These entities can be reached over web services using SOAP. The calls to the local representations are tunneled over the physical network to the Network Manager running on the target device, which delegates the call to the real entity.

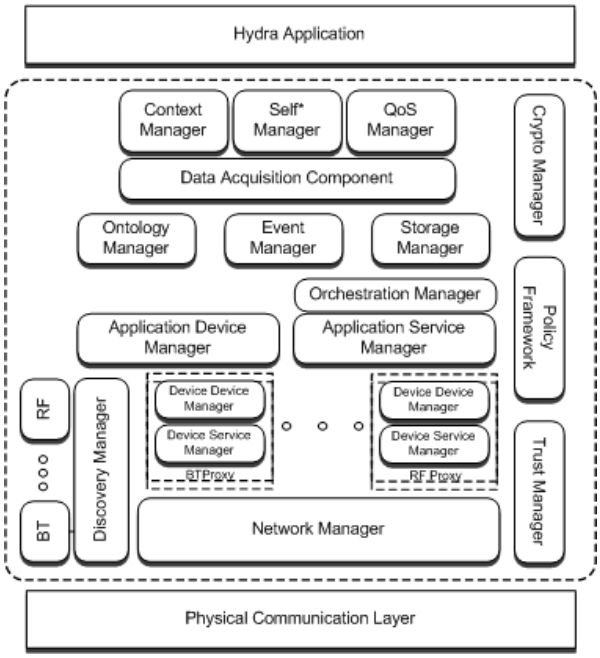


Figure 2: Hydra Managers

For communication each entity in the network becomes a number of Hydra IDs (HIDs). Each HID points to exactly one entity, and therefore the Network Manager can route data to the right target. Entities can have several HIDs for different users or contexts. Hydra enabled UPnP devices can get these HIDs by the Discovery Manager, which can find the devices in the local network and register them in the middleware.

The Crypto Manager allows building up a declarative security layer. Together with the Trust Manager and the Policy Frameworks it can protect the whole communication in Hydra. These Managers can also be used to build security in devices, as will be shown later for the storage area.

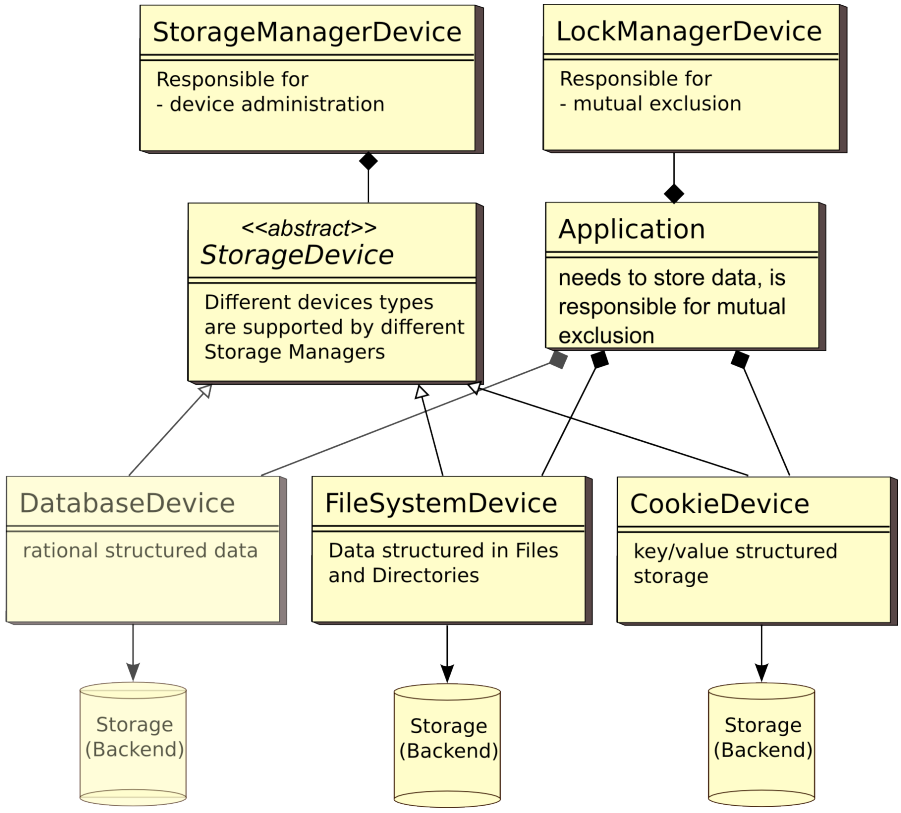


Figure 3: Hydra Storage Architecture

Storage is served in Hydra using the Hydra Storage Architecture described in Figure 3. Each device serving storage to the middleware has a Storage Manager running. This manager is responsible for administration of storage on the local device. The storage itself can be brought to the middleware as some kind of virtual device. As an example we implemented the Cookie Device and the File System Device.

The Cookie Device represents a key/value storage. The implemented version realizes this by using a hash table, which can be configured to be persistent or in memory.

A File System Device allows to access data in files and structured in directories. We implemented three kinds of File System Devices:

- Local File System Device: This device is developed to build a lightweight component to bring local storage into the Hydra middleware. Therefore it delegates the functions of a File System Device to a directory in a local file system.
- Striped File System Device: This device distributes its data over some backend devices without redundancy. Therefore it is something like a RAID 0 implementation.
- Replicated File System Device: This device stores its data on each of its backend devices. Therefore it is some kind of RAID 1 implementation.

The Striped and the Replicated File System Device are designed to exist in parallel on a number of physical devices. The File System Devices can be stacked to improve the value of the storage. Figure 4 shows a combination of Local, Striped and Replicated File System Devices.

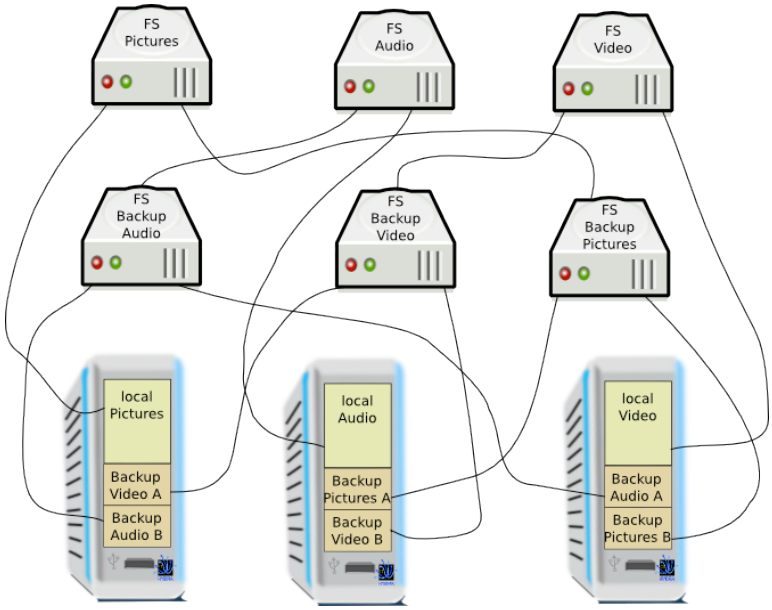


Figure 4: Example of combined File System Devices

In this example we have three Hydra enabled hard disks. Each disk holds an area of storage to hold local data and an area to keep backup data for the other two volumes. Holding the Striped File System Devices (FS Backup Audio, Video and Pictures) and the Replicated File System Devices (FS Audio, Video and Pictures) on each physical device it is possible to access the whole data even if one device is taken away. More it is also possible on a single device to access the local stored data using the Replicated File System Device. On reconnection the Replicated File System Devices can synchronize each other.

Security in the Hydra Storage Architecture is realized using the Security Architecture. The Crypto Manager together with the Trust Manager and the Policy Framework allow a secure communication over the underlying network. These Managers can also be used to encrypt data on local storage in a way, that only authorized users can access it. This can be realized independent of the File System Devices. Figure 5 shows an example of a protocol storing information in such a secure way, that even an administrator on all physical devices cannot read the data.

To realize this level of security the user creates a symmetric key for his data and stores it encrypted by his public key on the device. If other users shall also access the data, the key is also stored encrypted with their public key. All data is then stored using the symmetric key. To access the data it is now necessary to hold the symmetric key the data is encrypted with. This key can only be decrypted having the private key of an authorized user. To protect the data against changes it is possible to store a signature for the data on the device.

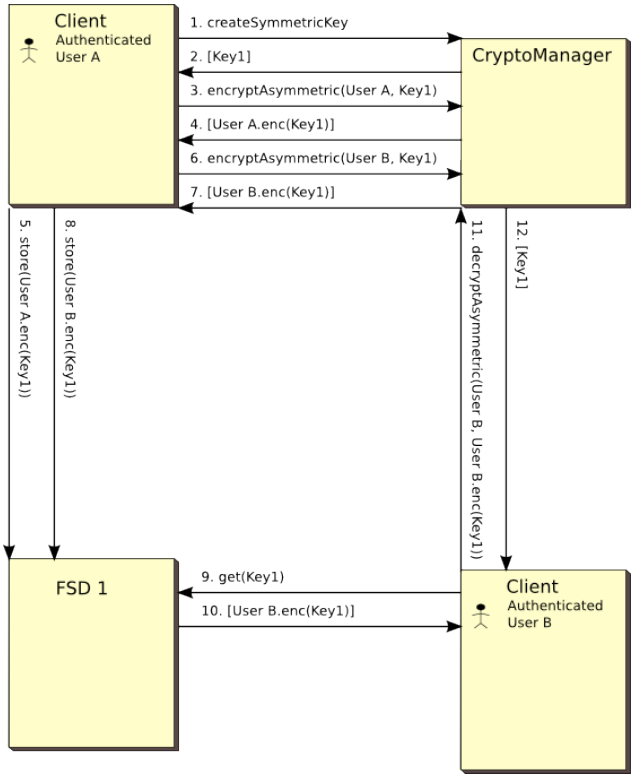


Figure 5: Data Encryption in Hydra Storage Architecture

A special role in the Hydra Storage Architecture has the Lock Manager. This Manager can be used to synchronize access to File System Devices. It is planned to give applications the ability to lock files and directories, so that an atomic access can be realized. The devices itself do not care about locking, this is lead to the application developer. This area will be the most important of the future work of the PC^2 in the Hydra project.

References

[1] Brinkmann, A; Effert, S., Gao, Y., Hansen, K. M., Kool, P.: D3.17 Final Storage Architecture Report, Hydra EU Deliverable, Germany, 2009

5.3 Distributed and parallel applications

5.3.1 Enabling Heterogeneous Hardware Acceleration Using Novel Programming and Scheduling Models (ENHANCE)

Project coordinator	Prof. Dr. André Brinkmann, University of Paderborn
Project members	Jun.-Prof. Dr. Christian Plessl, PC², University of Paderborn
	Tobias Beisel, PC², University of Paderborn
Supported by:	Federal Ministry of Education and Research (BMBF)

General Problem Description

ENHANCE is a research project carried out by German scientific as well as industrial partners. The project aims at a better integration and simplified usage of heterogeneous computing resources within current and upcoming computing systems.
Heterogeneous systems contain multiple compute components, like multi-core processors, complemented by graphics processing units (GPUs), and/or field programmable gate arrays (FPGAs). Employing such hardware architectures raises several challenges in programmability, performance estimation, and scheduling that are approached within the ENHANCE project and shall result in a framework enabling the development and use of applications on heterogeneous systems. The benefit of the developed methods for the industrial application partners is of special importance within the project.

The project is funded by the German Federal Ministry of Education and Research (BMBF) under grant number 01|H11004 and is scheduled with a runtime from April 2011 to September 2013. ENHANCE is one of twelve projects sponsored within the "HPC-Software für skalierbare Parallelrechner" program that aims at promoting the high performance computing (HPC) community in Germany. The Paderborn Center for Parallel Computing role within the project is a combination of research on topics of the project and managing the cooperation of the partners as the official coordinator of the consortium. The project's website can be found at [1].

Problem details and work done

Heterogeneous accelerator systems are commonly used since the appearance of multicore CPUs and general-purpose graphics processing units (GPUs). Multi-core CPUs combined with GPUs are even provided in standard configurations of personal computers.

Data centers also set up experimental cluster systems, which use a combination of multi-core processors, GPUs, and specialized co-processors, such as ClearSpeed CSX or FPGAs. This is particularly interesting for solving medium-sized scientific computing problems in a cost-effective and energy-efficient way without accessing large and expensive super computing systems. This enables nearly every scientific and industrial institution to access significant computing power to address increasingly complex numerical simulations.

At the current state of the art, the usage of such systems is still limited since most accelerators need to be programmed with proprietary and unfamiliar programming languages and Application Programming Interfaces (APIs). The efficient development of software for these architectures is highly challenging for inexperienced developers and requires knowledge about the underlying hardware and software components. Hence, the research effort within the ENHANCE project addresses the challenges to ease the development and use of hardware accelerated codes.

Detailed ideas of the project

Facing the first challenge of simplifying software development for heterogeneous systems, we intend to develop a tool-flow that automatically parallelizes loops within an application. We, therefore, aim at performing the following steps to automate the source-to-source code transformation:

- First, we use a source code parser to analyze the application and translate the code into a polyhedral representation.
- This intermediate representation may then be optimized by index transformations to maximize the number of independent indices.
- To map these optimized index paths to the available hardware architectures, we aim at extending the PLUTO project to support heterogeneous architectures. PLUTO already allows mapping a polyhedral model to OpenMP. The result will be an internal representation of the index paths for a specific architecture.
- To generate the code for the target architecture, we intend to use the CLooG tool and extend it to fit the needs of heterogeneous target languages.

In a second part of the project, we approach the challenge of performing scheduling decisions at runtime and treat hardware accelerators as peer computation units that are managed by the scheduler, like CPU cores. The goal of scheduling tasks in the context of heterogeneous systems is to assign tasks to compute units in order to enable time-sharing of accelerators and to provide fairness among tasks that compete for the same resources. We are, therefore, aiming to

- include specific hardware characteristics and the status of the available heterogeneous compute units
- obtain knowledge about the availability and suitability of a task for a particular hardware accelerator from the application
- introduce a scheduling and programming model that allows preemption and migration for accelerators
- evaluate scheduling policies to be used for the scheduling decision
- implement an extension of the Linux Completely Fair Scheduler to support heterogeneous computing components.

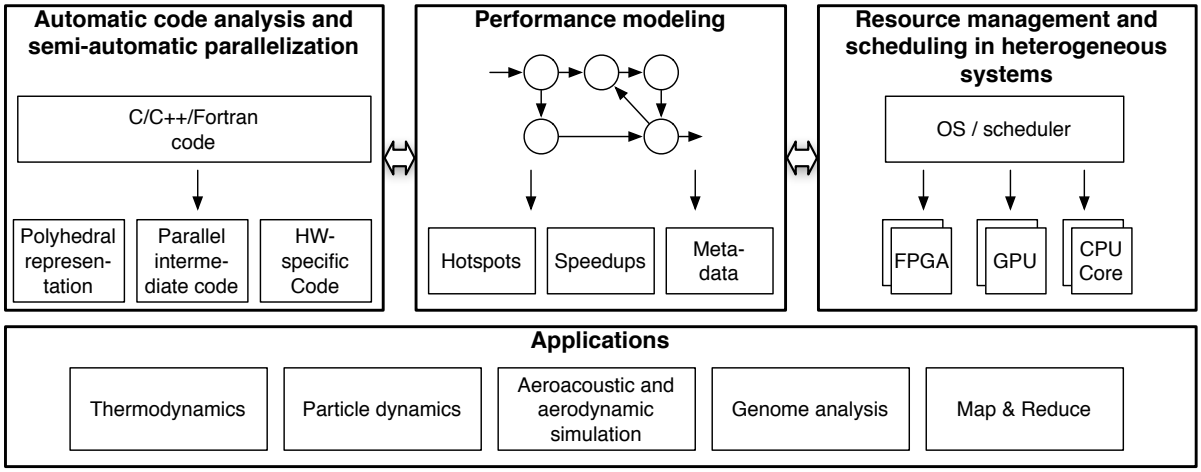


Figure 1: General structure of the ENHANCE project.

Both of these parts have a strong demand for profiling data to optimize their results and a need for well defined interfaces between application and operating system. In a third part of the ENHANCE project, we, therefore, intend to develop a performance model and a fat binary mechanism. The performance model shall

- precisely measure and provide runtimes, input- and output data volumes, and operation counts of prototype functions and later also generated function implementations using benchmarks tools
- provide a model to describe operations and dependencies

The intended fat binary model shall include

- a metadata-model for application parameters to be used for scheduling tasks
- binaries for all target architectures.

The partners aspire a complete framework (cf. Figure 1) allowing both automatic parallelization and scheduling of applications on heterogeneous hardware architectures and optimizing the results with an included performance model. During the process of developing the needed tool-flow and the operating system extension, we evaluate our intermediate results iteratively by incorporating the industrial partner's challenging applications on bio-informatics, automotive computing, pollutant dispersion, and thermodynamics.

Scheduling on heterogeneous systems

The Paderborn Center for Parallel Computing sets its main focus within the project on the development of a scheduler for the heterogeneous target systems. This includes developing a scheduling framework, designing and evaluating scheduling algorithms, and providing a fairly easy programming model to be used with the scheduler. Most of the work done so far was on the theoretical evaluation of a heterogeneous scheduling environment in Linux and the development of a prototype scheduler for a system combining a multi-core-CPU and a Graphics Processing Element (GPU).

Scheduling such heterogeneous systems is more difficult than traditional CPU scheduling due to several reasons:

1) Accelerators typically do not have autonomous access to the shared memory space of the CPU cores and an explicit communication of input data and results is re- quired. Furthermore, the communication bandwidth, latency, and performance characteristics of accelerators are non- uniform and need to be considered when mapping tasks. These characteristics also determine the granularity of the functionality that can be successfully scheduled without too much overhead (single operations, kernels, functions/library calls, threads).

2) Most accelerator architectures do not support preemption but assume a run-to-completion execution model. While computations on CPU cores can be easily preempted and resumed by reading and restoring well defined internal registers, most hardware accelerators do not even expose the complete internal state nor are they designed to be interrupted.

3) Heterogeneous computing resources have completely different architectures and ISAs. Hence, a dedicated binary is required for each combination of task and accelerator, which prevents the migration of tasks between arbitrary compute units. Even if a task with the same functionality is available for several architectures and if the internal state of the architecture is accessible, migrating a task between different architectures is far from trivial because the representation and interpretation of state is completely different.

Derived from these challenges, the major results of a first deliverable [2] can be described as follows: We propose the extension of the Linux Completely Fair Scheduler (CFS) to treat accelerators as peer computation units that may be used almost equally to the available CPUs. Extending the operating system to be aware of available compute units

and being able to use these as allocatable compute units in an extended CFS would allow scheduling decisions to not only incorporate the applications inputs but also the state of the accelerators and the operating system. This approach would imply using the scheduling model of the CFS that uses preemptive multitasking with time-sharing.
We propose to overcome the absence of preemption on some accelerators by introducing the use of cooperative multitasking with checkpoints to the programming and scheduling models. Tasks may voluntarily release accelerator architectures at application defined checkpoints and, thus, make it possible to be “preempted” and suspended at these points and, thus, to be enqueued again for later continuation or migrated to a different architecture.
As operating systems cannot directly handle “hardware threads”, i.e., the threads executed on the accelerator, we introduce so-called “delegate threads” that are a logical representation of their hardware counterparts running on the accelerators. These threads are in charge of performing all communication with the accelerator and the hardware thread execution. They, thus, handle all functionalities that have to be carried out by using the vendor specific APIs from the user space. These threads may then be scheduled with the normal Linux CPU scheduler and, therefore, also satisfy the Linux scheduler’s need to provide threads as schedulable entities.
To evaluate the CFS extension, we also propose to develop an additional user space scheduler that avoids the limitations of the kernel space approach by simplifying the use of the scheduler for the application but being less applicable for short-term inclusion of the system status and utilization. Nevertheless, this enables the pervasive analysis and comparison of the kernel space scheduler.

Based on these ideas, a first prototype scheduler and a corresponding programming model were implemented as an extension for Linux and presented in [3]. Its general architecture is shown in Figure 2.

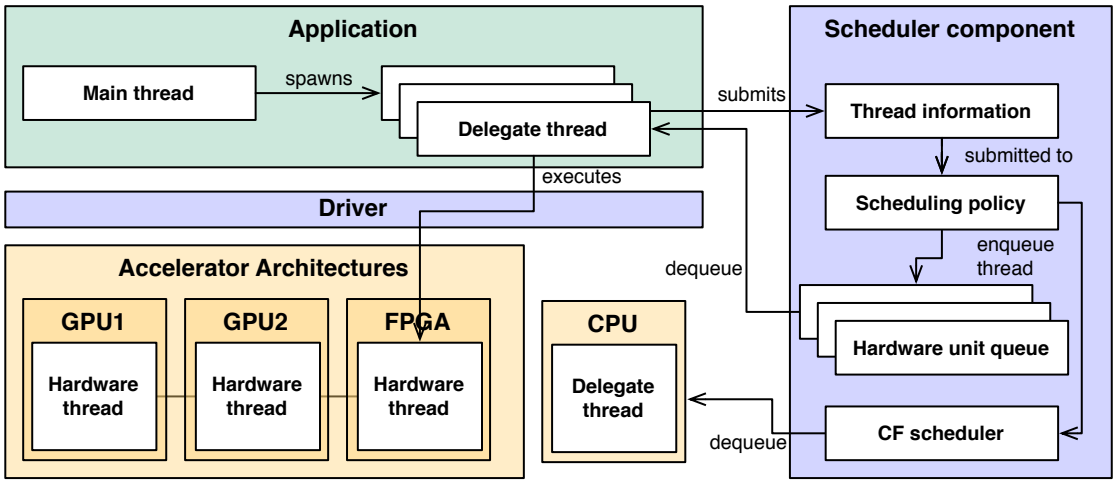


Figure 2: General scheduling model for heterogeneous systems.

Resource Usage

The funding for the project included the purchase of a GPU cluster system. Two nodes, each equipped with 2 INTEL XEON E5620 CPUs, 4 Nvidia C2070 GPUs, and 48 GB main memory, are provided by the PC² to the partners for the development and evaluation of parallel applications within the project. In addition local workstations were equipped with cutting edge GPUs to allow easy and fast development cycles.

References

[1] www.enhance-project.de

[2] Beisel, T.; Plessl, C. and Brinkmann, A.: Approaches towards managing heterogeneous Computing Resources in Linux, internal deliverable of the ENHANCE project, September 2011.

[3] Beisel, T.; Wiersema, T.; Plessl, C. and Brinkmann, A.: Cooperative Multitasking for Heterogeneous Accelerators in the Linux Completely Fair Scheduler, in Proceedings of 22nd IEEE International Conference Application-specific Systems Architectures and Processors (ASAP), 2011.

5.3.2 Domain Specific Approaches for the Acceleration of Computational Nanophotonics Simulations with CPUs, GPUs and FPGAs

Project coordinator	Jun.-Prof. Dr. Christian Plessl, PC^2, University of Paderborn
Project members	Björn Meyer, PC², University of Paderborn Dr. Jens Förstner, Department of Physics, University of Paderborn
Supported by	University of Paderborn Research Award 2009

General Problem Description

For many scientific domains, the subject of investigation, for example, financial markets, chemical molecules, or nanophotonic structures, the mathematical modeling of problems, has advanced to a level where the behavior of a system can be predicted with sufficient accuracy. However for many real-world problems, no analytical or statistical solutions, which could be approximated, are available; hence, numerical computer simulation is required to evaluate the system's properties or dynamics.

A common way to model optical phenomena in nano-structured material is to solve the Maxwell equations for a given problem. They can be approximated with sufficient accuracy by the Finite-Difference Time-Domain (FDTD) method [6, 7]. Therefore, we are able to model light-material interaction and control the light field for a given nano structure.

In fact, the complexity of the systems under consideration today requires massive computing power. State-of-the-art simulation models are so complex that the runtime of the simulation on traditional computers is becoming a limiting factor for the discovery of new scientific results [5]. Consequently, the provision of high compute power is the enabler for answering future research questions in the field of computational nanophotonics. The traditional way of addressing this high computational demand is to develop hand-optimized code for high-performance parallel computer clusters that distribute the calculations among different processors and computer nodes.

We study an alternative approach, which leverages recent advances in computer architecture to address the challenges of scientific computing, especially by considering new General Purpose Graphic Unit (GPGPU) architectures and other emerging parallel computer architectures, such as field-programmable gate arrays (FPGAs), floating-point accelerators, and Many-Core processors. To achieve this goal, we abstract the physical problem description from accelerator architecture details by providing a source-to-source code generator. Problem instances formulated in a Domain Specific Language (DSL) are translated into high-efficient accelerator specific code. Consequently, physicists can focus

on their models instead of investing time into complex accelerator architecture and programming paradigms.

Problem details and work done

In the following section of this report, we present the specifics of computational nanophotonics within the context of a common nano structure setup. Subsequently, we illustrate our code generation flow and first performance results for generated GPU code. The next-to-last section presents a Multi-Core and a FPGA accelerator prototype implementation.

Computational Nanophotonics – a real world simulation

In this work, we investigate accelerating simulations from the domain of computational nanophotonics although our code generator is not limited to this kind of simulations. As in many other simulations, partial differential equations (PDEs) are used in nanophotonics to model a physical phenomenon. These PDEs are solved by nearest neighbor computation on a finite regular grid. Algorithms that share these properties are commonly referred to as stencil algorithms.

One frequently used nano structure for testing and benchmarking purposes is the microdisk cavity in a perfect metallic environment. Both, the numerical stability and the well-known simulation result in the form of whispering gallery modes, are the reason for its attractiveness. The supporting material of the setup is ideal metal, while the microdisk contains a vacuum. At the beginning of the simulation, all fields are initialized to zero and only one discrete grid point in space is used to excite the E- and H-fields. In the first time steps of the simulation, a damped oscillation sine wave in z-direction is excited. Due to the damping, the source is negligible after a few simulation steps and only the dispersion of the fields must be computed. Typically, we want to visualize the energy density by computing the point-wise summed square of the H-field in z-direction during the detection time period for each point on the two dimensional discrete grid. The result of such an energy density computation is shown in Figure 1.

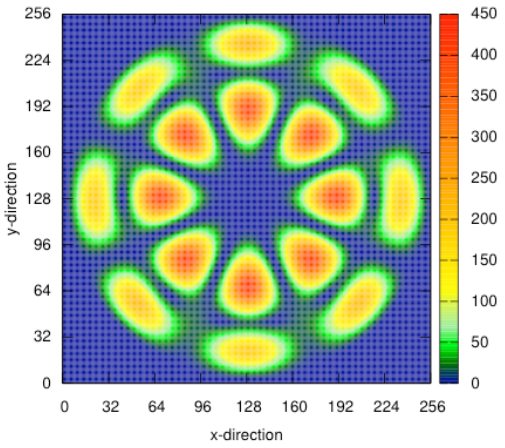


Figure 1: Time-integrated energy density for a microdisk cavity in a perfect metallic environment

Code Generator Toolflow

Aiming at code generation for stencil computations, we implemented our own code generator from scratch to meet the specialties of stencil computation. We assure that our approach is applicable to a wide range of stencil computations by introducing several novel concepts, which are incorporated into our DSL definition. For real-world stencil computations, update equations might be different for regions with different properties (e.g., different materials in nano structures). Furthermore, we must consider special cases at material or spatial borders. We address these special needs by introducing the concept of spatial domains and subdomains. We can assign properties to both domain types and update equations (or sequences of them).

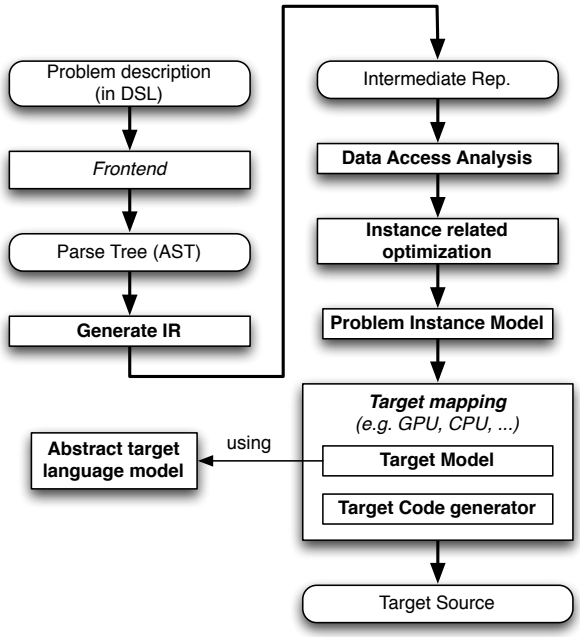


Figure 2: Code generation toolflow

Stencil computations formulated in our DSL are parsed by our Code Generator front end. The update equations and corresponding domains and subdomains are transformed into an intermediate representation (IR). In a successive step, we analyze equations that are parsed into an Abstract Syntax Tree (AST) and perform a data access analysis. Inside our code generation flow, we have a model for the problem instance and an abstract language model, which allows us to easily integrate new code generation back-ends for novel accelerator architectures.

Currently, we are able to generate code for single- and multi-GPU workstations as well as for GPU clusters (nodes with multiple GPUs are supported). For those back-ends, we have implemented several code generation and optimization options. By using our code generator, we achieve a 35.9x speed-up with two GPUs compared to single Core Performance [1].

Prototypes for Multi-Cores and Convey HC1 (FPGA based) accelerators

To support Multi-Cores and FPGAs targets with our code generator, we evaluate those architectures with computational nanophotonics simulation prototypes.

For FDTD, the number of arithmetic operations per point in space is small compared to the number of needed memory accesses for those computations. Therefore, we focus mainly

on optimizing the usage of the memory hierarchy of Multi-Cores and the Convey HC-1 computer.

A common technique to achieve reuse in Multi-Core memory hierarchies is spatial tiling. This implies that we partition the setup into 2D tiles for the microdisk cavity in a perfect metallic environment. On NUMA systems, we require a memory region (a tile) to be updated by the same core each time. Nevertheless, because of the FDTD nearest neighbor computation, the borders of the tiles will be accessed by cores, which are updating the neighbor tiles. Compared to naïve parallel OpenMP implementations, we get a speed-up of up to 1.8x on an eight-core machine with our prototype implementation.

The Convey HC-1 is a server system with a novel architecture. Standard components, like the Intel Xeon CPU, are supplemented with a coprocessor. This coprocessor consists of multiple individual, programmable FPGAs. In this manner, application developers can utilize the FPGAs to accelerate a certain class of applications. A developed FPGA hardware description is called Personality in the context of the Hybrid-Core architecture. To design custom Personalities for a specific application, one can use the Convey Personality Development Kit (PDK). However, it is not mandatory to develop a Personality; instead one or more of the provided Personalities can be used. In this work, we investigate the performance of the provided Vector Personalities, which are not restricted to a special application domain [2, 3, 4]. The Development effort for using vector personalities is similar to the OpenMP parallelization effort because the application developer only needs to place pragmas in the source code. However, it is mandatory for high-performance to support the compiler with the stencil application and the architectural details of the Convey computer in mind by placing special operations in the source code.

Personalities are running on the Application Engines of the Convey coprocessor and access the dedicated memory with a maximal bandwidth of up to 80 GB/sec. For the coprocessor memory, we can use two different kinds of memories standard DIMMs, which are optimized for 64 Byte accesses (optimized to cache lines of modern processors) and convey a scatter gather DIMM, which is optimized for 8-Byte accesses.

When comparing the performance of the Multi-Core- (tiled OpenMP) and the Convey-implementation (standard DIMMs installed), we get a comparable speed-up. Using the scatter gather RAM, the speed-up increases by 1.8x to 2.2x, which illustrates the importance of the effective usage of the memory interface for bandwidth limited stencil computations.

Resource Usage

Development and parameter studies of FDTD GPU code was done on several workstations equipped with NVIDIA HPC computing cards (Tesla C2050, C2075). Multi Core and NUMA implementations of FDTD were developed on workstations hosted by the PC^2. The Convey HC-1 was used to evaluate the vector personalities of the Convey computer, which is also hosted by the PC^2.

References

[1] Meyer, B.; Plessl, C. and Förstner, J.: „Transformation of Scientific Algorithms to Parallel Computing Code: Single GPU and MPI multi GPU Backends with Subdomain Support.“ In *Symposium on Application Accelerators in High Performance Computing (SAAHPC)*, 2011. Knoxville, Tennessee, USA, Juli 2011. IEEE.

[2] Augustin, W.; Weiss, J.P. and Heuveline, V.: Convey HC-1 hybrid core computer – the potential of FPGAs in numerical simulation. In Prof. Int. Workshop on High-Performance and Hardware-aware Computing, Karlsruhe, Germany, Mar. 2011. KIT Scientific Publishing.

[3] Bakos, J.: High-performance heterogeneous computing with the convey hc-1. Computing in Science Engineering, 12(6):80–87, Nov–Dec 2010.

[4] Brewer, T.: Instruction set innovations for the convey hc-1 computer. Micro, IEEE, 30(2):70–79, 2010.

[5] Dineen, C.; Förstner, J.; Zakharian, A.; Moloney, J. and Koch, S.: Electromagnetic field structure and normal mode coupling in photonic crystal nanocavities. Opt. Express, 13(13):4980–4985, June 2005.

[6] Taflove, A. and Hagness, S.: Computational electrodynamics: the finite-difference time-domain method. Artech House antennas and propagation library. Artech House, 2005.

[7] Yee, K.: Numerical solution of inital boundary value problems involving maxwell’s equations in isotropic media. IEEE Transactions on Antennas and Propagation, 14:302 – 307, May 1966.

5.3.3 Medical Image Processing

Project coordinator	Prof. Dr. Marco Platzner, University of Paderborn
Project members	Tobias Beisel, PC², University of Paderborn
Supported by:	Zentrales Innovationsprogramm Mittelstand (ZIM)

General Problem Description

Magnet Resonance Imaging (MRI) is an important part of current internal medicine, allowing non-invasive examinations to diagnose and treat medical conditions. It is used especially to identify and classify cancer and tumors before applying surgical procedures.

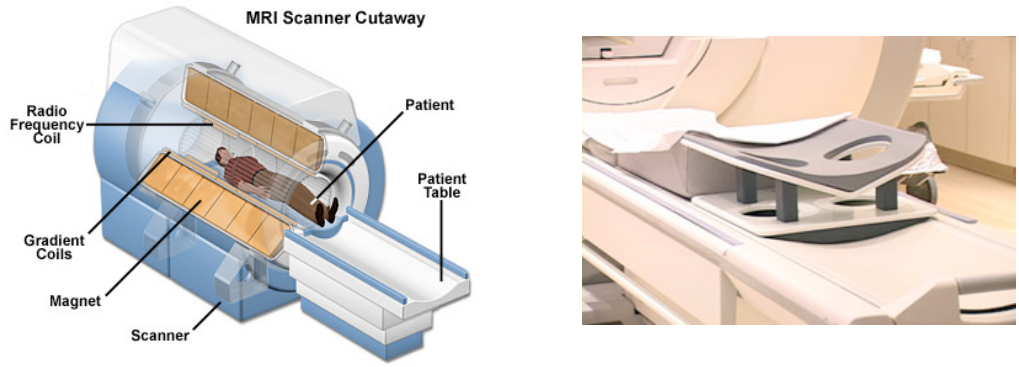


Figure 1: Setup of a MRI system and a breast MRI attachment

MRI uses a powerful magnetic field and radio frequency during the measurement and a computer to reconstruct detailed images of bones, soft tissues, and organs from the measurements. Figure 1 shows the general setup of an MRI scanner.

Passing an electric current through the wire loops surrounding the patient creates a strong magnetic field. In the meantime, the radio frequency coils in the magnet send and receive radio waves. This triggers protons in the body to align themselves. Once aligned, radio waves are absorbed by the protons, which stimulate spinning. Energy is released after exciting the molecules, which then emit energy signals that are picked up by the coil. This information is sent to a computer, which processes all the signals and generates an image. The final product is a 3-dimensional image representation of the examined area. Unlike Computed Tomography (CT) scanning or general x-ray studies, no ionizing radiation is involved in an MRI. One major field of application is breast cancer and tumor detection, using an attachment to the scanner as shown in Figure 1. An exemplary outcome is shown in Figure 2, where cancer can be detected in the upper half of the right breast (white area). Subsequent to the measurement, a tool-flow of different image processing algorithms is used to support the radiologists in identifying and rating tumor and cancer cells. These

include image registration, image segmentation, and different methods of feature extraction and classification. The registration aims at matching a series of 3D images measured over time such that they can be used as a single image for diagnosis purposes. The segmentation additionally extracts special regions of interest from the images so that irrelevant information are deleted. This might, for example, emphasize the tumor regions or the affected organs. Finally, a classification uses feature extraction algorithms to analyze the images based on special medical knowledge of the nature of tumor or cancer regions.

All of these algorithms work for large amounts of data and experience a long runtime based on complex algorithms. Waiting for the results involves a burden for the patient and reduced cost efficiency for the medical institute. The CADMEI GmbH [1] and the PC² aimed at developing new algorithms and finding customized solutions to accelerate the application. The main approach was to parallelize the algorithms, using different current and new parallelization techniques on diverse parallel hardware. For this purpose, a hardware was chosen that best suited the set of algorithms. Thus, a combination of different parallel architectures was incorporated to speed up the complete toolflow. These architectures especially include current Many-Core architectures and Graphic Processor Units (GPUs) but also more specialized and less common Field Programmable Gate Arrays (FPGAs).

Problem details and work done

The aforementioned algorithms were designed, developed, and accelerated in a ZIM [2] funded cooperation between the PC² and the CADMEI GmbH. The project was funded for 2 years and executed between July 2009 and June 2011. The goal of the project was to develop a fast and fully automatic diagnosis tool for female breast cancer detection, using MRI. The developed system implements the complete tool-flow needed to evaluate already reconstructed images for a fully qualified diagnosis. Beforehand, existing systems demanded a high level of interaction with the radiologist and delivered only low-level computer aided detection methods. Replacing a second radiologist demands high-quality results. The developed system provides algorithms of high-level knowledge representation, like automatic segmentation, and, thus, enables an automatic morphological rating of tracer accumulations.

In addition, it is in the interest of the patient and the medic to have the examination's findings available as soon as possible. Thus, acceleration using parallel architectures was incorporated. While the CADMEI GmbH was responsible for algorithm and framework development of the diagnosis system based on medical knowledge, the role of the PC², during the algorithm development, was to consult the CADMEI GmbH with the design of data structures, which map very well on parallel architectures. Moreover, the PC² ported

parts of the developed algorithms on appropriate architectures and designed a hardware product setup that is optimized in terms of a cost-benefit ratio.

Image Registration

In most cases an MRI examination delivers a series of images over time. To use the overall information of these images, they have to be combined into a single image before further analyses can be done. Different registration algorithms solve this requirement, but only very few available solutions support a 3-dimensional registration.

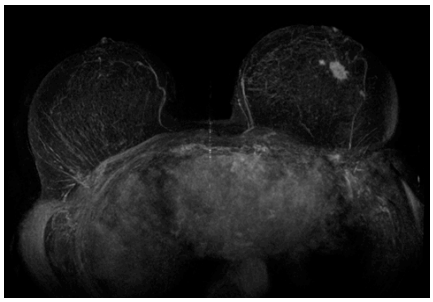


Figure 2: Resulting image of a breast MRI examination using the developed toolbox.

We provide a 3D registration algorithm that is capable of being extended for the use of a multi-modal registration. The algorithm uses a 3D extended version of the KLT Feature Tracker [3]. Feature Tracking is one of the fundamental operations in image processing. Tracking algorithms were originally developed to track features in movie sequences. Features are recognizable characteristics of the images, typically corner-like structures. Surveys have been made about what features are used best [4]. As movie sequences at bottom are a series of images, this technique can be transferred to the medical image registration. Extracting and tracking features from a series of MRI images provides sufficient information on how subsequent images are situated so that they can be morphed into each other by translation vectors. Figure 2 shows the result of a registration using the newly developed algorithms.

Image registration can be divided into several sub-algorithms that are executed sequentially. These sub-algorithms were subject to parallelization within the project and were ported to GPUs and Multi-Core architectures.
The Feature Search algorithm is done on a pixel-basis and, thus, could be performed independently for each of the pixels so the GPU was best suited for this part. Afterwards, each feature could be tracked in parallel and the morphing of the images could be done in parallel as well. While the tracking was best suited for the CPU because of the individual handling of each feature in the tracking algorithm, the morphing was optimally suited for the GPU as the same operations are performed on all pixels.

Image Segmentation

Using the registered images, the segmentation detects those regions that are most probably cancer or tumor regions. They are detected by the color value of each voxel. The voxels with the highest values are presumed to be situated in malicious regions. The most common segmentation algorithms use these voxels as starting points for region growing methods. They let the detected regions grow until the voxel values fall below a certain threshold. Choosing this threshold carefully is essential for good results. This is trained by learning algorithms, using examples with well-known results. Figure 3 gives an example of segmented regions in a scan (see ROI).

In this part of the project, a new algorithm was developed, which is based on complex mathematical calculations. This new approach required new techniques of acceleration and was implemented on a FPGA. The method is subject to non-disclosure-agreements and cannot be described in detail.

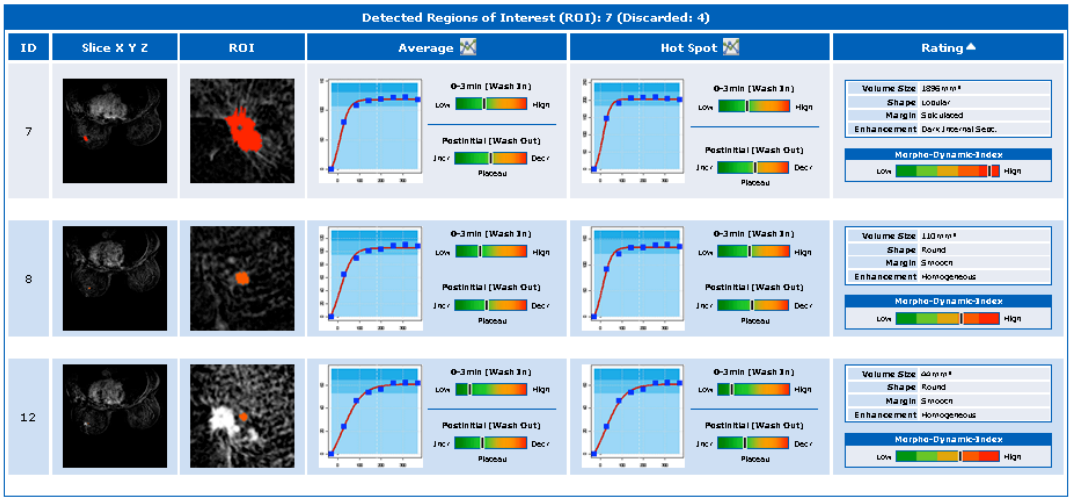


Figure 3: Example report after classification of the images

Feature Extraction and Classification

Classification algorithms predict the dignity of the segmented objects, rating if they are good-natured or malicious. Also, false-positive extracted objects have to be eliminated. The used algorithms are standard algorithms, which do not have to be accelerated.

Feature extraction algorithms play a major role in preparing the classification and delivering a characterization of segmented objects, which allow a rating of suspicious regions. Feature extraction is based on a list of feature descriptions, which can be extracted by image processing algorithms independent of the image quality. The descriptions specify well-known medical characteristics of malicious cells. The runtime of this step depends on the number of segmented objects and the length of the feature list. The better the intended result, the longer the list of features and the longer the needed runtime. CPU parallel execution was best suited because of the number of different

algorithms to be applied. The results are presented to the radiologist in a report (cf. Figure 3), which is used to verify the radiologists's traditional diagnosis. Using the report as a secondary diagnosis makes for a fast feedback to the patient.

Diagnosis and Performance Results

Table 1 compares the CPU speedups to the system existing prior to the project. Speedups of up to 14.6 were achieved using only CPU parallelization.

Algorithm	**Speedup factor (CPU only)**
Registration	2.1
Segmentation	14.6
Feature extraction	13.7
Classification	1.4

Table 1: Speedups compared to the 2009 system (only CPU parallelization)

Table 2 shows an overview of the runtimes achieved using accelerators.

Algorithm	**Average (sec.)**	**Min (sec.)**	**Max (sec.)**
Registration (CPU)	253	202	437
*Feature Search (CPU)	8	5	12
*Feature Search (GPU)	4	2	8
*Feature Tracking (CPU, per series)	16	15	19
*Interpolation (per series)	<1	<1	<1
*Morphing (CPU, per series)	43	29	75
*Morphing (GPU, per series)	8	2	18
Segmentation (CPU, seeding algorithm)	38	5	237
*Seeding algorithm	38	5	237
*Segmentation algorithm (FPGA)	10	4	34
*Segmentation algorithm (GPU)	1045	176	4342
Feature extraction	46	8	156
Classification	5	5	5
Total (only CPU)	342	220	835
Total (with GPU)	303	190	774

Table 2: Runtimes of the newly developed systems, using different technologies

Regarding an achieved prediction rate of 89% sensitivity and 76% specificity, the results show that a full diagnosis of high quality could be computed in an average of about 5 minutes - 13 minutes in the worst case - compared to up to several hours before.

Resource Usage

Within this project a set of different up-to-date accelerator technologies was used. A current NVIDIA GeForce GTX 580 GPU and 8-core Workstations were used as well as an Altera Stratix II FPGA.

References

[1] CADMEI – Software für Medizinsysteme GmbH, Website: www.cadmei.com
[2] Zentrales Innovationsprogramm Mittelstand, Website: http://www.zim-bmwi.de
[3] Tomasi, C. and Kanade, T.: Detection and Tracking of Point Features, in CMU Techreport, 1991
[4] Shi, J. and Tomasi, C.: Good features to track, in Proceedings of Computer Vision and Pattern Recognition, IEEE, 1994

5.3.4 Massively Parallel Monte-Carlo Tree Search

Project coordinator	Prof. Dr. Marco Platzner, University of Paderborn
Project members	Dr. habil. Ulf Lorenz, University of Darmstadt
	Lars Schäfers, PC², University of Paderborn
Supported by:	Microsoft Research, Cambridge

General Problem Description

Since the world's best chess players are computers as of some years ago, computer scientists are now faced with the next challenge, which is the ancient Asian board game called Go (chin.: wéiqí, kor.: baduk)[1]. Computing the best - or at least a good - move to a given board position for the game of Go is the focus of researchers working on artificial intelligence since many years. While applying the same algorithms and ideas that were used for the chess game showed only little success, a Monte-Carlo based approach brought about great improvements in the past few years [2]. The best computer-Go programs have recently reached a good amateur playing strength.

In the game of Go, the number of possible moves a player can select from is very large compared to the playable moves in the chess game. Furthermore, there are no good strategies known to diminish the number of moves in a save way by classifying, for example, senseless or clearly bad moves. In addition, static evaluation functions that can approximate the value of intermediate game positions could not be found even after extensive research for many years. As a result, we need to explore a very large game tree to compute a good follow up move to a given game situation. In a Monte-Carlo (MC) based approach, only single paths of the whole tree are explored, and statistics about the outcomes are collected on tree nodes close to the root. We call this exploration of a single path from the root node to a leaf node of the game tree a MC simulation. The accuracy of the statistics and, therefore, the accuracy of the whole procedure increases with the number of samples done. We focus on the parallelization of MC Tree Search (MCTS) algorithms for shared and distributed memory machines in order to increase the possible number of computable simulations per time unit.

Problem details and work done

MCTS is a simulation-based search method that brought about great success in the past few years, regarding the evaluation of stochastic and deterministic two-player games. MCTS learns a value function for game states by consecutive simulation of complete

games of self-play, using randomized strategies to select moves for either player. Especially in the field of Computer Go, an Asian two-player board game, MCTS surpasses traditional methods such as alpha-beta search [3]. MCTS may be classified as a sequential best-first search algorithm [4], where "sequential" indicates that simulations are not independent of each other as is often the case with MC algorithms. Instead, statistics of past simulation results are used to help future simulations find the most promising path in the search space's in a best-first manner. This dependency and the need to store and share the statistics among all computation entities makes parallelization of MCTS for distributed memory environments a highly challenging task. Parallelization of traditional search is a pretty well solved problem, e.g., see [5,6]. While it is sufficient for the alpha-beta search to map the actual move stack onto memory, MCTS requires us to keep a consecutively growing search tree representation in memory. On SMP machines, sharing a single search tree representation in memory is straight-forward and has already been proven to be very effective for MCTS parallelization [7,8]. However, sharing a search tree as the central data structure in a distributed memory environment is rather involved and only few approaches have been investigated so far [9]. We investigated a novel approach for the parallelization of MCTS for distributed high-performance computing (HPC) systems. Our algorithm spreads a single search tree representation among all compute nodes (CNs) and guides simulations across CN boundaries, using message passing. We map search tree nodes to randomized hash values and the hash values to CNs in an equally distributed fashion, which makes spreading tree nodes a straight-forward procedure. A comparable approach with traditional alpha-beta search was termed transposition driven scheduling (TDS) [10]. Computing more simulations in parallel than cores are available allows us to overlap communication times with additional simulations. We evaluated the performance of our parallelization technique on a real-world application. Our high-end Go engine Gomorra. Gomorra has proven its strength at the Computer Olympiad 2010 in Kanazawa, Japan, and just recently at the Computer Olympiad 2011 in Tilburg, Netherlands.
For further information about our work and the algorithms used, we refer to our according papers [11,12].

Resource Usage

To carry out our research, we made extensive use of the ARMINIUS+ cluster and the SMP-Server.
The ARMINIUS+ cluster was used to empirically determine the scalability of our parallel MCTS algorithm. It uses message passing to communicate between CNs and shared memory for inter-thread communication on the CNs. Multithreading is performed using POSIX-threads, while message passing is realized with OpenMPI. The variety of provided compilers and MPI distributions allowed us to get the best out of the available HPC systems. It turned out that the cluster management software CCS that is used with the

ARMINIUS+ cluster showed great flexibility, which was necessary to reserve compute resources in fixed time slots. This allowed us to play tournaments with Go-engines of other research groups around the world under hard time-constraints, using the internet. Apart from this, we used ARMINIUS+ to run several hundred sequential instances of our program at a time for extensive parameter studies. The SMP server was used for machine learning algorithms (because of the large amount of available main memory) and shared-memory scalability experiments (because of the large amount of available cores).

References

[1] see http://www.intergofed.org/ (website of The International Go Federation) for more information.
[2] Gelly, S.: et al. Modification of UCT with Patterns in Monte-Carlo Go. Technical Report 6062, INRIA, 2006.
[3] Knuth, E. Donald and Moore, Ronald W.: An Analysis of Alpha-Beta Pruning. In Artificial Intelligence, volume 6, pages 293-327. North-Holland Publishing Company, 1975.
[4] [Silver, D.: Reinforcement Learning and Simulation-Based Search in Computer Go. PhD thesis, University of Alberta, 2009.
[5] Donninger, C.; Kure, A. and Lorenz, U.: Parallel Brutus: The First Distributed, FPGA Accelerated Chess Program. In 18th International Parallel and Distributed Processing Symposium. IEEE Computer Society, April 2004.
[6] Himstedt, K.; Lorenz, U. and Möller, D.: A Twofold Distributed Game-Tree Search Approach Using Interconnected Clusters. In Euro-Par, volume 5168 of LNCS, pages 587-598. Springer, 2008.
[7] Chaslot, C.; Winands, M. and Jaap van den Herik,H.: Parallel Monte-Carlo Tree Search. In Conference on Computers and Games, pages
[8] 60-71, 2008.
[9] Enzenberger, M. and Müller, M.: A Lock-free Multithreaded Monte-Carlo Tree Search. In 12th International Conference on Advances in Computer Games, volume 6048 of LNCS, pages 14-20. Springer-Verlag, May 2009.
[10] Amine Bourki et al. Scalability and Parallelization of Monte-Carlo Tree Search. In International Conference on Computers and Games, pages 48-58, 2010.
[11] John W. Romein et al. Transposition Table Driven Work Scheduling in Distributed Search. In National Conference on Artificial Intelligence, pages 725-731, 1999.
[12] Schaefers, L.; Platzner, M. and Lorenz, U.: UCT-Treesplit - Parallel MCTS on Distributed Memory. MCTS Workshop, Freiburg, Germany, June 2011.
[13] Graf, T.; Lorenz, U.; Platzner, M. and Schaefers, L.: Parallel Monte-Carlo Tree Search for HPC Systems. In Proceedings of the 17th International Conference, Euro-Par 2011, Bordeaux, France, August/September 2011. LNCS, vol. 6853, pp. 365-376. Springer, Heidelberg.

5.3.5 SCALUS – On the Impact of Randomized Data Distribution Strategies on Storage Systems

Project coordinator	Prof. Dr. André Brinkmann, Johannes-Gutenberg-University of Mainz
Project members	Ivan Popov, PC², University of Paderborn
Supported by:	The EC FP7 Marie Curie Initial Training Networks (ITN) SCALUS – SCAling by means of Ubiquitos Storage under Grant Agreement no. 238808

General Problem Description

Storage research increasingly gains importance based on the tremendous need for storage capacity and I/O performance. Over the past years, several trends have considerably changed the design of storage systems, starting from new storage media over the widespread use of storage area networks, up to grid and cloud storage concepts. Furthermore, to achieve cost efficiency, storage systems are increasingly assembled from commodity components. Thus, we are in the middle of an evolution towards a new storage architecture made of many decentralized commodity components with increased processing and communication capabilities, which requires the introduction of new concepts to benefit from the resulting architectural opportunities.

The vision of the MCITN SCALUS is to deliver the foundation for ubiquitous storage systems, which can be scaled in arbitrary directions (capacity, performance, distance, security). Providing ubiquitous storage will become a major demand for future IT systems and leadership in this area can have significant impact on European competitiveness in IT technology. To get this leadership, it is necessary to invest into storage education and research and to bridge the current gap between local storage, cluster storage, grid storage, and cloud storage.

Storage systems typically use clustered architectures designed to store and process this information efficiently. An important aspect of building scalable storage systems is the underlying data distribution strategy. A variety of randomized solutions handling data placement issues have been proposed and utilized. However, to the best of our knowledge, there has not yet been a structured analysis of the influence of pseudo random number generators (PRNGs) on the data distribution. The result of this analysis helps to choose a PRNG according to the quality of the load distribution and the performance, but has been very computational and memory intensive.

Problem details and work done

With the fast growth of digital information, storage systems are becoming larger and are having to store ever increasing amounts of data. They are also faced with more stringent requirements on their characteristics, such as throughput and response time. One of the main challenges is to distribute data across many storage devices. The more evenly the data is distributed, the more load-balanced the storage system is, whereas unbalanced systems entail a risk of bottlenecks at the most heavily loaded storage nodes.

Data stored in storage systems can be distinguished to be either metadata or user data, where metadata is information about the data. Storage technology evolved from traditional file storage systems, with metadata and data managed by the same machine, to storage systems where metadata and user data are separated. This scheme, consisting of client, metadata and storage nodes, solves some scaling issues by providing clients with the ability to access directly the stored data after getting information about their location from metadata nodes. Nevertheless, it now becomes necessary to scale both, metadata servers and storage nodes.

The exascale territory, which is currently being approached by the high-performance computing (HPC) and storage communities, makes necessary systems with thousands of storage nodes and hundreds of metadata nodes. These clusters require techniques to distribute data across their nodes most efficiently in terms of load balancing (quality of distribution) and performance (distribution time).

Randomization can help to improve adaptability in the presence of changing sets of storage nodes. These approaches are typically implemented by using hash functions, which are based on pseudo-random number generators (PNRGs). One of the first approaches to randomly distribute data blocks for storage systems has been proposed by Korst. Random duplicated assignment stores multimedia data by assigning a number of copies of each data block to different, randomly chosen disks, where the number of copies may depend on the popularity of the corresponding data [Korst97].

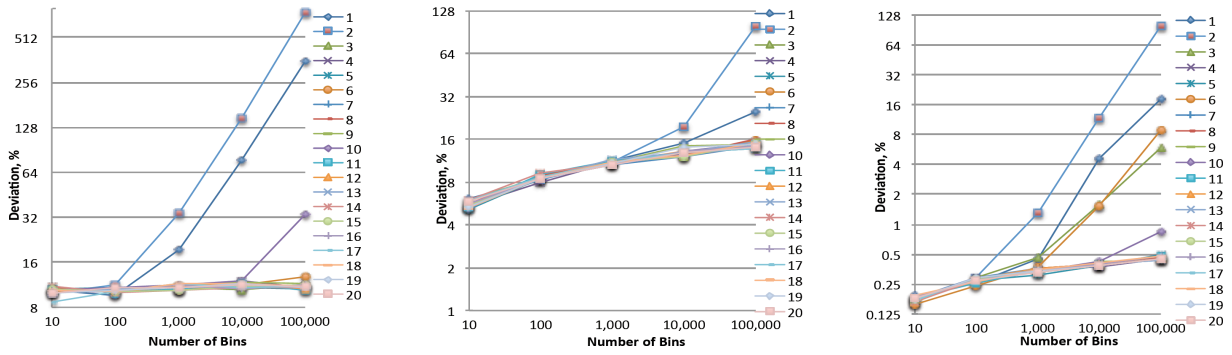
Karger et al. presented an adaptive hashing strategy based on randomization for homogeneous settings that satisfies load-balancing properties and adaptivity [KLL+97]. Their Consistent Hashing strategy randomly maps storage systems and data items to a [0;1) ring. The mapping between data items and storage systems is performed by assigning a data item to the unique storage system that is mapped “in front” (w.r.t. position on the ring) of the data item. Nevertheless, the deviation from a fair data distribution can be very high if each storage system is represented by a single point on the ring. Therefore, Karger et al. introduce the concept of virtual storage systems, so that each storage system is represented by a set of virtual storage systems on the ring.

We analyze different PRNGs in two different settings (please see Table 1 for list of evaluated PRNGs) [PBF12]. In the first part we consider Consistent Hashing [KLL+97] as a combination of two consecutive phases: bins’ distribution and balls’ distribution. The second part evaluates a data distribution strategy proposed by the authors of this paper in previous work, which uses many more random experiments and therefore depends more heavily on the calculation time needed for each random experiment [BEMS07][MEK+11].

Table 1: Investigated PRNGs

Nr.	PRNG	Source	Type
1	minstd_rand	Boost library	LC
2	minstd_rand0	Boost library	LC
3	hellekalek1995	Boost library	inversive LC
4	rand48	Boost library	library
5	ecuyer1988	Boost library	additive combine (LC +LC)
6	kreutzer1986	Boost library	suffel output (LC improved)
7	taus88	Boost library	XOR LFS
8	ranlux3	Boost library	swc + discard block
9	ranlux64_3	Boost library	swc + discard block
10	ranlux3_01	Boost library	swc + discard block
11	ranlux64_3_01	Boost library	swc + discard block
12	mt11213b	Boost library	Twisted GFSR
13	mt19937	Boost library	Twisted GFSR
14	Lagged_fibbonacci_607	Boost library	Lagged fibonacci
15	Unix rand()	Unix implementation	MC
16	SHA-1	Gcrypt	SHA-1
17	MD-5	Gcrypt	MD-5
18	Tiger-192	Gcrypt	Tiger
19	RMD-160	Gcrypt	RMD
20	Whirlpool	Gcrypt	Whirlpool

The following small subset of results outlines the impact of the PRNGs on the quality of the data distribution for Consistent Hashing in different settings, especially scaling the number of storage devices, resp. bins, *n* and the number of data items, resp. balls, *m* (see [PBF12] for the complete analysis):



(a) Bins' distribution (b) Balls' distribution (m/n = 1,000) (c) Balls' distribution (m/n = 1,000,000)

Figure 1: Influence of PRNGs on bin's load. Maximum deviation per bin.

The maximum deviation of the bins, as shown in Figure 1(a), has a big impact on the usable capacity of a storage system. It can be seen that minstd_rand, minstd_rand0 and ranlux3_01 have a very poor scaling behavior. The results for the other PRNGs are close to each other, but fluctuations of 4% to 5% can still have a huge impact on the usable capacity of a device. Furthermore, all of them have a maximum deviation of around 10% throughout all tests. This limits the usable capacity for all devices if no other means to overcome deviations are in use.

According to Raab and Steger [RS98] the maximum load of any bin is $m/n+\alpha*(2\ m/n*\log_2 n)^{1/2}$, i.e. of the order of the mean plus some deviation when $m >> n*\log_2 n$. Our experimental results match this theoretical statement with high precision. Calculating the maximum deviation with this formula gives a result very close to our experimental 16% (see Figure 1(b)).

Considering the maximum deviation per bin, a graphic of the same form as for m/n = 1,000 case can be noticed as well (see Figure 1(c)). The maximum deviation from the graphic with value of 0.5% again matches with Raab and Steger theorem.

The results of this investigation reveal the influence of PRNGs on different data placement strategies for storage systems. This influence is dependent on many different factors, such as the data distribution strategy, the number of balls, and the number of bins. Therefore the more of them are taken into account while choosing a PRNG, the more proper a choice can be made in terms of best characteristics. The deviation from the average bin load using Consistent Hashing can be decreased two- or threefold by increasing either the number of bins or the balls-to-bins ratio 1.000 times. In practice this leads to the choice between increasing the number of servers that store the data and their substitution by new ones with bigger capacities.

A main PRNGs' characteristic that should be considered during their choice is performance. Even those PRNGs having similar quality of distribution metrics can differ in performance up to 29 times. One of the first applications for the provided results above can be in the direction of QoS. Given information about performance and distribution accuracy of PRNGs, one can manipulate data distribution in storage systems in order to provide certain QoS guarantees.

References

[1] Brinkmann, A.; Effert, S., Meyer auf der Heide, F. and Scheideler, C.: „Dynamic and Redundant Data Placement“. In Proceedings of the 27th IEEE International Conference on Distributed Computing Systems (ICDCS), 2007

[2] Karger, D.R., Lehman, E.; Leighton, F.T.; Panigrahy, R.; Levine, M.S. and Lewin, D.: “Consistent hashing and random trees: Distributed caching protocols for relieving hot spots on the world wide web”. In Proceedings of the 29th ACM Symposium on Theory of Computing (STOC), 1997, pp. 654–663.

[3] Korst, J. H. M.: “Random duplicated assignment: An alternative to striping in video servers”. In Proceedings of the 5th ACM International Conference on Multimedia (Multimedia), 1997, pp. 219–226.

[4] Miranda, A.; Effert, S., Kang, Y., Miller, E.L.; Brinkmann, A and Cortes, T.: “Reliable and randomized data distribution strategies for large scale storage systems”. In Proceedings of the 18th International Conference on High Performance Computing (HiPC), 2011

[5] Popov, I.; Brinkmann, A. and Friedetzky, T.: “On the Influence of PRNGs on Data Distribution”. In Proceedings of the 20th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP), 2012.

[6] Raab, M. and Steger, A.: ”Balls into bins” - a simple and tight analysis”. In Proceedings of the Randomization and Approximation Techniques in Computer Science, Second International Workshop (RANDOM), 1998.

5.4 Testbeds and Benchmarking

5.4.1 System Evaluation, Benchmarking and Operation of Experimental Cluster Systems

Project coordinator	Dr. Jens Simon, PC², University of Paderborn
Project members	Axel Keller, PC², University of Paderborn
	Andreas Krawinkel, PC², University of Paderborn
	Holger Nitsche, PC², University of Paderborn
Supported by:	Fujitsu Technology Solutions
	ict AG

General Problem Description

In the year 2009, the PC² has installed an InfiniBand connected cluster system for the research groups of the theoretical physics of the University of Paderborn. The system consists of 57 compute nodes with 456 processor cores and 1560 GByte of main memory. A Network Attached Storage (NAS) system is connected with 1-Gigabit-Ethernet to all nodes of the cluster. The capacity of harddisks of the NAS is 48 TByte. Emerging technologies, computer systems, interconnects, and software systems have been evaluated by the PC² in the selection phase of the cluster system and further evaluations are done for the next generation systems. Besides system evaluation and benchmarking of new cluster technologies, different experimental or special purpose cluster systems are operated for research groups of the University of Paderborn.

Problem Details and Work Done

Different computer systems and cluster technologies have been evaluated. The tested systems are up-to date two sockets Intel Xeon systems with dual- and quad-core processors, two and four sockets AMD Opteron with dual- and quad-, and hexa-core processors, and some special purpose computer systems with reconfigurable hardware. These systems were equipped with different configurations of high-speed interconnects (InfiniBand single, double, and quad data rate, MyriNet 10G, and Gigabit Ethernet) and different operating systems of Linux and the Microsoft operating system Windows HPC Server 2008. All benchmarking results are published on the web sides of the Paderborn Benchmarking Center (1).

Co-Operations: The PC² benchmarking center is also doing system evaluation and benchmarking for external companies and organizations. The PC² has a long-term co-operation with Fujitsu-Siemens Computers where Paderborn acts as a Competence Center for High Performance Computing. System benchmarking is also done for the companies ict AG and christmann informationstechnik + medien.

Company	Provided Equipment
ict AG	Loan equipment 4 two sockets quad-core Xeon systems 7/09 - 9/09
BlueArc	NAS Titan T2200, 30TByte 08/09 – 09/09
NetApp	NAS FAS 3170, 8TByte 08/09 – 09/09
Isilon	NAS IQ 9000, 27 TByte 08/09 – 10/09

References

[1] Simon, J.: PC² Benchmarking Center, http://wwwcs.uni-paderborn.de/pc2/about-us/staff/jens-simons-pages/benchmarkingcenter.html

5.4.2 Onelab2: OneLab Extensions Towards Routing-in-a-Slice

Project coordinator	Prof. Dr. Holger Karl, PC², University of Paderborn
Project members	Jens Lischka, PC², University of Paderborn
Supported by:	7th Framework of the European Commission

General Problem Description

Alongside networking research, experimentally-driven research is key to success in exploring the possible futures of the Internet. In PlanetLab Europe, the OneLab project provides an open, general-purpose, and shared experimental facility, both large-scale and sustainable, which will allow European industry and academia to innovate today and assess the performance of their solutions.

The second phase of the OneLab project, OneLab2, builds on the original OneLab project's foundations, continuing work on the PlanetLab Europe testbed, increasing its international visibility and extending it in both functionality and scale.

PlanetLab was originally built to develop new technologies for distributed storage, network mapping, peer-to-peer systems, distributed hash tables, and query processing. To do so researchers built their own overlay network topologies in user space. There was no need for direct Layer2 access or for the creation of Layer2 topologies. As a consequence the evaluation and testing of new, IP-independent routing protocols (e.g. data centric networking, pub/sub systems) on PlanetLab nodes is a problem, as it is not possible for a process running in PlanetLab to distinguish between different incoming interfaces or to determine which outgoing interface to use -- the very core function of a router cannot be emulated. The cause of this problem is PlanetLab's network virtualization design.
Routing-in-a-slice (RiaS) tries to overcome this problem by the application of new network virtualisation techniques on the PlanetLab Europe infrastructure. The objective is to offer researchers a convenient tool for building their own, custom, virtual Layer2 topologies on PlanetLab Europe, upon which it becomes meaningful to execute lower-level routing and forwarding experiments.

Problem Details and Work Done

Layer2 topology creation on PlanetLab nodes has to deal with three main issues:

1. Network Virtualisation,
2. Virtual Network Mapping (VNM), and
3. Monitoring.

Network virtualisation is necessary to realize to run many virtual networks with different topologies, each of them running its own protocol, routing software etc., upon the same network infrastructure simultaneously without affecting each other. Currently it is impossible to create topologies that behave as if they were actual Layer2 topologies on the current PlanetLab platform.

Another aspect of virtual topology creation is the Virtual Network Mapping problem. Once a researcher has defined his custom virtual network topology in a *Network Topology Description File*, the components (virtual nodes and links) have to be mapped onto the physical nodes and links (or possibly even paths) of the underlying infrastructure. This task is called *virtual network mapping (VNM)* or Resource Mapping. This job is done by the *VN Mapper*.

In contrast to existing Resource Allocation approaches like SWORD [1] or NetFinder [2] our RiaS system not only chooses a set of appropriate nodes that satisfy the researchers needs but is also able to create the desired topology.

Finally, monitoring is needed to feed the VN Mapper with a proper description of the available resources (e.g. CPU for nodes and data rate for links) of the physical network. This information can be collected inside a central database; in a later implementation, it need not be centralized but can also be distributed.

RiaS Architecture

Figure 1 depicts how the components of our RiaS system work together. A central database holds information about the available resources of the PlanetLab Europe network and creates the physical network topology description. This physical network information is accessible to the VN Mapper, either upon request or by periodic updates.

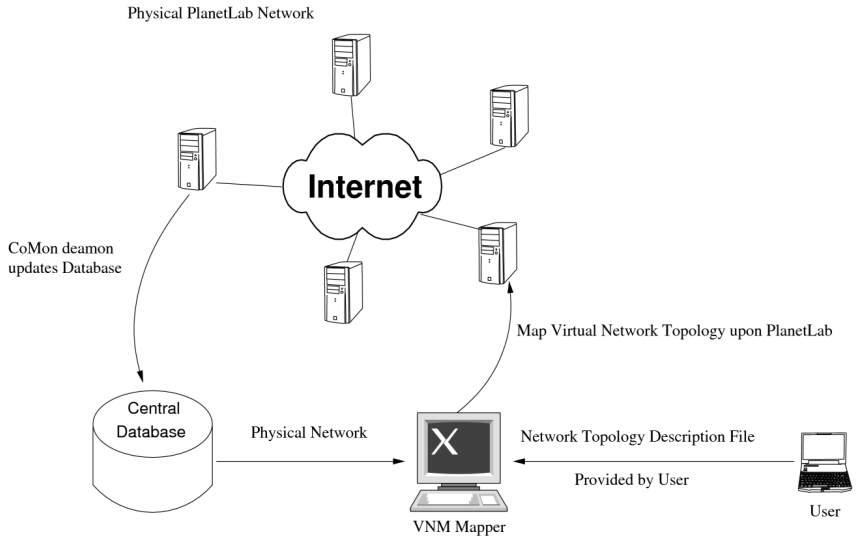


Figure 1: RiaS Architecture

A researcher who wants to reserve a slice to run a network experiment sends a description of his custom network topology, a so called *virtual network requests (VNR)*, to the VN Mapper. The VN Mapper then tries to allocate the necessary PlanetLab resources and configures the topology. Finally, the researcher is informed when and where his slice is available and can login to the nodes of the requested topology to run his tests.

Extension Requirements

Our objective is to enable PlanetLab to be used as evaluation and test platform for new routing protocols that do not rely on IP. In addition researchers should be able to run their existing routing software and protocols on PlanetLab without the need to modify them. To achieve this routing functionality we must be able to create and configure multiple network interfaces inside PlanetLab slices.

In the following section we provide answers to the questions how the network virtualisation on PlanetLab currently works, what kind of problems arise related to routing-in-a-slice, and how we can fix this.

Network Virtualisation

PlanetLab uses the Linux VServer technology [3] for host virtualization. Linux VServer is a container-based virtualization approach which allows several virtual Linux hosts to run simultaneously on a single, shared kernel; no virtual host has direct access to the hardware. A set of such virtual hosts working together forms a PlanetLab slice. This concept allows to share the hardware resources of PlanetLab nodes among a large number of PlanetLab slices simultaneously in a very efficient way.

This container-based virtualisation design has also a disadvantage. All VServer virtual hosts share the same kernel and also share the same network stack. The issue is that routing experiments have to manipulate the central routing table and that such a routing

table manipulation would affect all other experiments running on this PlanetLab node since they all share the same kernel and in particular the same network stack.
Suppose one Planetlab slice wants to change his default route in the routing table. This would change the default route for all remaining PlanetLab slices on this particular node since they all use the same routing table, leading to great chaos. As another example, suppose a user wants to add a virtual network interface. This interface would be visible and configurable by all other slices on this PlanetLab node since it is added to the shared stack which is accessible by all slices.
PlanetLab solves this problem by a very restrictive VServer network configuration setting. The VServers on PlanetLab nodes are configured in such a manner that the user owns no rights to change/add routing table entries or to configure/add new NICs or tunnels. This makes it hard for researchers to build their own topologies on PlanetLab. In particular it is impossible to build Layer2 topologies.
Currently the network virtualisation on PlanetLab is done by *PlanetLab Virtualised Network Access (VNET) [4]*. VNET relies on *Connection Tracking* which is part of Linux's *Netfilter* system [5]. VNET associates every inbound packet with a connection structure which ensures that slices send and receive only packets associated with connections that they own. A connection structure mainly consists of source and destination IP address, source and destination port number, and an exchange ID (XID). Each time a connection is established such a structure is inserted into a connection table. The kernel now knows the connections of a particular slice and drops packets from other slices. Note that the distinction between connections on a PlanetLab node is actually done by port numbers and XID since all slices on a PlanetLab node share the same IP address. One major drawback with respect to our objectives is that VNET can only support the IP protocols TCP, UDP, ICMP, GRE and PPTP.
Although there exists a possibility to create packet sockets in VNET, called *Safe Raw Sockets*, their protocol family attribute must be set to PF-INET and thus their use is restricted to the IP protocol family and so there is no Layer2 access possible. This is a major problem for testing new IP-independent routing protocols or software on PlanetLab.
Thus, to create Layer2 topologies we have to make some changes to the virtualisation techniques that are currently in use. To overcome the problem RiaS makes use of container-based virtualisation approach extended by *Network Namespaces (NetNS) [6].* Network namespaces allow to assign a private set of network resources to one or several processes. These have their own set of network devices, IP addresses, routes, sockets, and so on. Other processes outside the namespace cannot access these network resources.
Using network namespaces on PlanetLab would require some changes to the PlanetLab kernel to integrate the NetNS patch set and run each VServer with its own network namespace, but it would not conflict with PlanetLab's container-based design philosophy and is therefore much better to migrate to PlanetLab compared to full virtualisation.

Virtual Network Mapping

A researcher should be able to associate the resources (nodes and links) of his virtual network topology with various capacity requirements (e.g. CPU, data rate, or delay) which must be satisfied by the underlying physical infrastructure.

Efficiently assigning virtual network resources to physical resources (Virtual Network Mapping) satisfying a previously defined set of capacity constraints is no trivial task and can be shown to be NP-complete. In addition to efficiency, a network mapping algorithm has to meet the following requirements:

- Make efficient use of the underlying physical resources such that a large number of virtual networks can be mapped onto the physical resources at the same time.
- Mapping virtual networks of reasonable size (100 - 200 nodes) should not take more than a few seconds.
- Handle dynamically arriving virtual network requests that stay in the network for an arbitrary time before departing.
- Handle admission control. Since the physical resources are limited, some virtual network requests have to be rejected or postponed to avoid violation of resource guarantees for existing virtual networks.

An existing approach that meets nearly all requirements is described in [7], but our evaluations showed that the algorithm is inefficient for larger network requests (virtual networks with more than 20 nodes). Therefore, we implemented our own virtual network mapping algorithm based on subgraph isomorphism detection which we presented at the VISA 09 workshop in August 2009 [8].

Monitoring

Monitoring the PlanetLab nodes is necessary to provide the VN Mapper with informations about the current resource consumption on the physical network. Currently we obtain node usage informations by periodically polling the *CoMon* [9] daemon on each PlanetLab node.

To measure the data rate, bandwidth and delay between PlanetLab nodes we use the results of the Scalable Sensing Service S^3 project [10] which is already deployed on PlanetLab. S^3 provides web-services based access to data-rate- and latency informations between all pairs of PlanetLab nodes.

References

[1] Oppenheimer, D.; Albrecht, J.; Patterson, D. and Vahdat, A.: Distributed Resource Discovery on PlanetLab with SWORD, In WORLDS, 2004

[2] Zhu, Z. and Ammar, M.: Overlay network assignment in PlanetLab with NetFinder, Technical Report GT-CSS-06-11, 2006

[3] Linux-VServer, http://linux-vserver.org
[4] Huang, M.: VNET: PlanetLab Virtualized Network Access, 2005
[5] Linux Netfilter, http://www.netfilter.org
[6] NetNS, http://lxc.sourceforge.net/network.php
[7] Yu, M.; Yi, Y.; Rexford, J. and Chiang, M.: Rethinking Virtual Network Embedding: Substrate Support for Path Splitting and Migration, SIGCOMM Comput. Commun. Rev., 2008
[8] Lischka, J. and Karl, H.: A virtual network mapping algorithm based on subgraph isomorphism detection, VISA '09: Proceedings of the 1st ACM workshop on Virtualized infrastructure systems and architectures, 2009
[9] Park, K. and Pai, V.: CoMon: a mostly-scalable monitoring system for PlanetLab, SIGOPS Oper. Syst. Rev., 2006
[10] Yalagandula, P.; Sharma, P.; Banerjee, S.; Basu, S. and Lee, S.J.: S3: A scalable Sensing Service for Monitoring Large Networked Systems, INM '06: Proceedings of the 2006 SIGCOMM workshop on Internet network management, 2006

6 User Projects

6.1 Simulation of Mass Transfer at Free Fluid Interfaces

Project coordinator	Prof. Dr.-Ing. Eugeny Kenig, University of Paderborn
Project members	Dr. Arijit Ganguli, University of Paderborn

General Problem Description

Mass transfer represents a central phenomenon in numerous processes of chemical and reaction engineering; very often it predetermines both the functionality and the quality of separation units and reactors. Consequently, the understanding and adequate modeling of mass transfer phenomena is extremely important. In systems involving fluid phases, e.g., in distillation, absorption, and extraction, mass transfer usually occurs between them, whereas the species are transported within the phases and across the phase's interface. Moreover, the interface itself is not fixed in space and time. It moves – often in a complex way - together with the contacting fluid phases and, hence, is called free or moving interface. Examples of such phenomena are given by species separation in absorption or distillation (with the interface between liquid films and a gas stream or between gas bubbles and a liquid stream) or in liquid-liquid extraction (with the interface between single droplets or droplet swarms and a continuous liquid phase). A number of other examples are available, and they can also be met in fluid reacting systems.

The description of the local interfacial mass transfer phenomena in a two-phase system results in a coupled problem that comprises momentum and mass transport at and around moving interfaces. The solution of such problems is very intricate because the movement of the interface influences interfacial mass transfer and vice versa Therefore, the traditional assumption that the velocity field is not affected by the concentration field cannot be made (two-way coupling). The complexity of this problem has been one of the major obstacles for the creation of rigorous first-principle-based models of mass transfer processes in real separation equipment units.

In past decades, enormous progress has been made in the development of powerful computers. This development significantly pushed forward numerical methods, among others, in the field of fluid dynamics. The efficient symbiosis of modern numerics and computer power has brought about the new emerging area of Computational Fluid Dynamics (CFD). Various problems that scientists and engineers did not dare to approach some twenty years ago have been successfully solved nowadays, using a wide range of

in-house and commercial software tools. This remarkable feature allows many important problems to be reconsidered with respect to their solvability and gives a fresh impetus to the investigation of mass transfer problems in fluid systems.
In our work, we focus on the modeling of mass transfer phenomena in multiphase fluid systems with a separated flow of phases. This means that, unlike in the distributed multiphase systems treated as interpenetrating continua, the interface is explicitly considered as an apparent boundary between continuous fluids.

Problem details and work done

We recently proposed a method capable of handling the interface deformation, the interfacial concentration jump, and the distribution coefficient variation [1,2]. The main idea of the present approach is to incorporate the interfacial boundary conditions (i.e., the continuity of the interfacial fluxes and the concentration jump at the interface) into the mass transfer equations themselves. This has to be done in such a way that the boundary conditions are only fulfilled in the region very close to the interface, whereas, outside this region, the mass transfer equations are valid.
More exactly, the interfacial mass transfer related boundary condition is directly implemented into mass transfer equations as an additional source term. Since two interfacial boundary conditions have to be fulfilled, mass transfer equations for both contacting phases should be solved for the whole computational domain. Therefore, the following extended equations are obtained:

$$\frac{\partial C_1}{\partial t} + \underset{\mathrm{r}}{u} \cdot \nabla C_1 = \nabla \cdot \left(D_1 \nabla C_1 \right) + \alpha \left(D_2 \frac{\partial C_2}{\partial n} - D_1 \frac{\partial C_1}{\partial n} \right)$$

$$\frac{\partial C_2}{\partial t} + \underset{\mathrm{r}}{u} \cdot \nabla C_2 = \nabla \cdot \left(D_2 \nabla C_2 \right) + \beta \left(\frac{C_2}{K_D} - C_1 \right)$$

At the interface, the values of and are set high enough (e.g., 10^4) so that the boundary conditions are only fulfilled there. In the rest of the computational domain, and are equal to zero, thus, transforming the equations into the common mass transfer equation.

The suggested method can be used for any fluid-fluid systems and permits to vary throughout the computational domain. Furthermore, the method is not limited to systems with binary mass transfer; it can readily be extended to multicomponent systems. In [1], a first implementation of the new method is demonstrated for a toluene-acetone-water

extraction system comprising a droplet rising in a continuous phase. The moving interface is described with the level set method. To solve the equation system, the commercial finite-element-based solver COMSOL Multiphysics 3.5a by COMSOL AB is applied. An example of the concentration field evolution during the droplet rise is illustrated in Figure 1. A decrease in acetone concentration inside the droplet can be observed. It starts from the centre (Figure 1a). As the droplet rises, the acetone flux from the droplet to the continuous phase results in the acetone traces around the droplet interface that are the most pronounced in the bottom part (Figures. 1b and 1c). An interplay of the convective flow and diffusion in the continuous phase brings about a streak of acetone directed downwards from the droplet bottom (Figure. 1d). The acetone concentration inside the droplet keeps on decreasing with time, whereas the streak follows the droplet rise path (Figures. 1e-1f). These simulation results are in a satisfactory qualitative agreement with the numerical results of Wang et al. [3]. Recently, a similar investigation was performed in [2] for a gas-liquid system, and the results were in line with the experimental observations by Kück et al. [4]. However, further work is necessary to reach the robustness and to investigate the applicability to multicomponent systems.

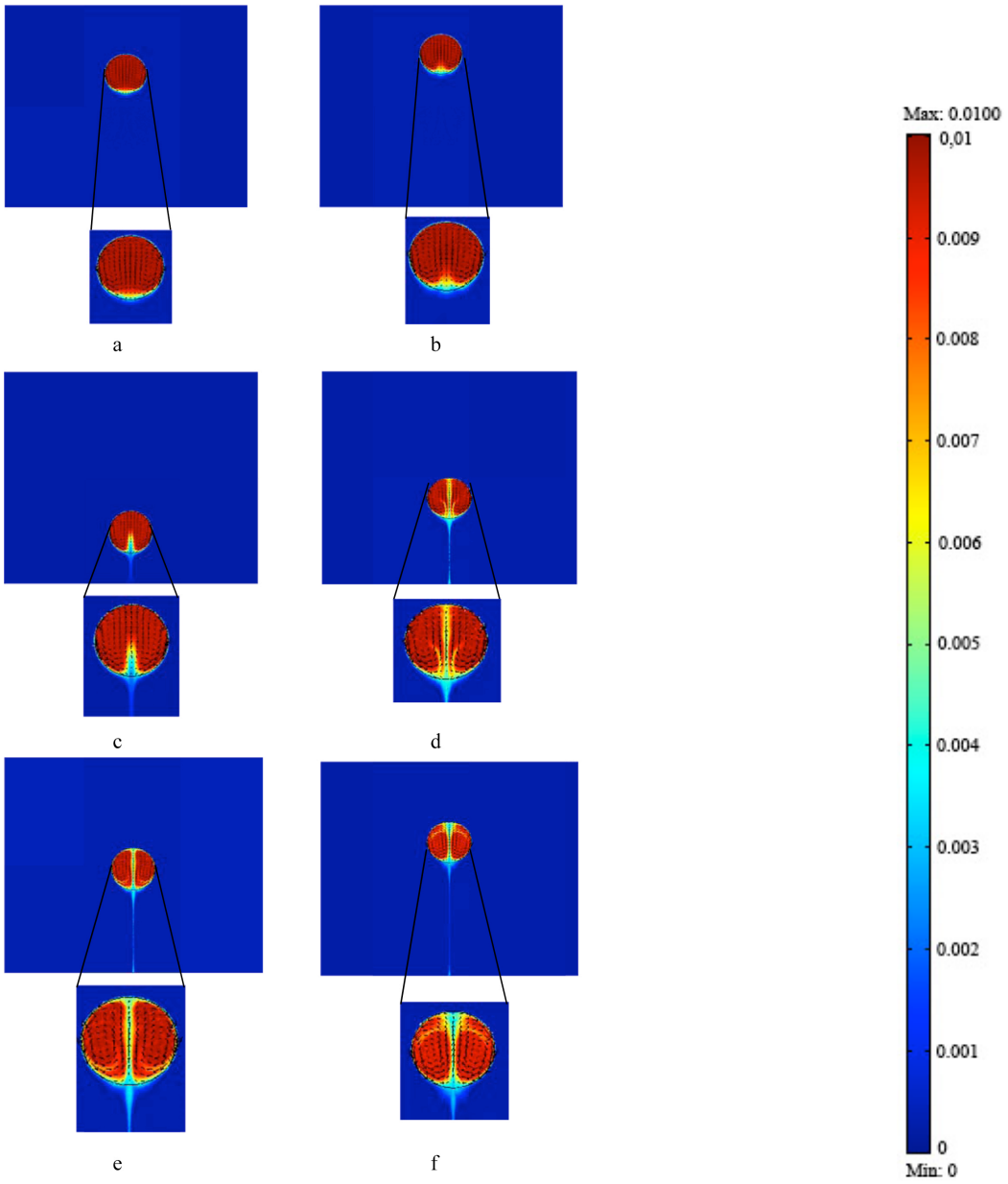


Figure1: *Concentration contours of acetone (gram/litre) and relative velocity vectors for a system with K_D = 0.63: t = 100ms (a); 120ms (b); 200ms (c); 500ms (d); 750ms (e); 1000ms (f).*

Resource Usage

CFD simulations of moving interfaces require significant computational power. Simulations are run on several nodes (typically 6 to 12) and usually take several days or even weeks. Consequently, we use the Arminius cluster every day, which is, therefore, an essential basis to our work. We use the commercial CFD-codes Star-CCM and COMSOL as well as the open source code OpenFOAM.

References

[1] Kenig, E. Y.; Ganguli, A.; Atmakidis, T. and Chasanis, P.: A novel method to capture mass transfer phenomena at free fluid-fluid interfaces. Chem. Eng. Process 50 (2011), 68-76.

[2] Ganguli, A. and Kenig, E. Y.: A CFD-based approach to the interfacial mass transfer at free gas-liquid interfaces. Chem. Eng. Sci. 66 (2011), 3301-3308.

[3] Wang, J.; Lua, P.; Wang, Z.; Yang, C. and Mao, Z.-S.: Numerical simulation of unsteady mass transfer by the level set method. Chem. Eng. Sci. 63 (2008), 3141-3151.

[4] Kück, U.D.; Schlüter, M. and Räbiger, N.: Analyse des grenzschichtnahen Stofftransports an frei aufsteigenden Gasblasen. Chem. Ing. Tech. 81 (2009), 1599-1606.

6.2 Simulation of crack propagation in functionally graded materials

Project coordinator	Prof. Dr. rer.nat. Maria Specovius-Neugebauer, University of Kassel Prof. Dr.-Ing. Hans Albert Richard, Universtiy of Paderborn
Project members	Dr. rer. nat. Martin Steigemann, University of Kassel Dipl.-Ing. Britta Schramm, University of Paderborn
Supported by:	Sonderforschungsbereich TR/TRR 30, DFG

General Problem Description

This work is part of the project “Risswachstum in funktional gradierten Materialien” within the collaborate research center SFB TR/TRR 30, kindly supported by the DFG. The main purpose of the project is the prediction and simulation of crack propagation processes especially in inhomogeneous materials.

Cracks in structures and components do not only limit their lifetime but may also cause catastrophic failures harming men as well as the environment. In the last decades, many efforts have been taken to understand crack growth and its predictability. However, the precise simulation of crack growth is still problematic. Defects or cracks in structural components can be caused by many different factors, even as early as the production process. Especially for safety aspects, an essential question is if a crack can be detected, and if not, how much this crack will grow till the next service. Moreover, can it become critical? New developments in material sciences and the growing application of anisotropic and also functionally graded materials to fulfill the more and more specialized demands on structural components in modern engineering have given an impulse to the study of fracture mechanics in such structures.

From a physical point of view, the energy principle, already formulated by Griffith in 1921, can also be used for the prediction of quasi-static crack propagation in anisotropic and inhomogeneous materials: A crack can only grow if there is enough energy to break the material and form a new crack front [4].

Based on the combined approach of mathematical modeling of the energy principle, experimental investigations, and numerical simulations, the main focus of this project is to get a deeper understanding of fracture processes in inhomogeneous (functionally graded) materials (FGMs).

Problem details and work done

The point of departure for modeling crack propagation was a mathematical asymptotic representation for the change of energy. As shown in [1], the change of energy caused by a crack elongation of length τ, directed at an angle θ in a plane two-dimensional linear elastic structure can be calculated to

$$\Delta\Pi(\theta) = 2\gamma(\theta)\tau - \frac{1}{2}\sum_{i,j=I,II} K_i M_{i,j}(\theta) K_j \tau + \mathcal{O}\left(\tau^{3/2}\right)$$

Here K_i are the stress intensity factors (SIFs) arising from the asymptotic representation of the displacement field near the crack tip. $M_{i,j}(\theta)$ are so-called local integral characteristics depending on the material properties and the shape of the crack elongation. The function $\gamma(\theta)$ represents the surface energy depending on the direction and following the energy principle that a crack can only grow if the change of total energy $\Delta\Pi(\theta)$ is negative for some angle θ and small τ. As part of this project, this representation was generalized to also cover anisotropic inhomogeneous (functionally graded) materials [9]. With a formula for the energy release rate at hand, it is possible to simulate the propagation of the crack step-wise [8]. In each simulation step, the direction of the crack can be determined by formula if all the SIFs and the quantities $M_{i,j}(\theta)$ can be computed numerically. For this reason, a framework for the numerical simulation of crack growth in plane solids was developed, called “MCrack2D”. MCrack2D is a pure research code based on finite elements with the intention to realize an exact-as-possible transfer of analytical models to numerics in order to test and improve theoretical ideas and make them finally applicable to real-world problems. Today, MCrack2D can handle anisotropic and inhomogeneous two-dimensional solids.

The key for the high-accurate computation of SIFs is an integral representation, which is a generalization of the ideas in the early works of Bueckner (1970) [3] and Mazya/Plamenevsky (1977) [5]:

$$K_j(u) = \int_{\partial G} \sigma^{(n)}(u) \cdot V^j - u \cdot \sigma^{(n)}(V^j)\, ds, \qquad V^j \sim r^{-1/2}$$

Here, u is the displacement field, $\sigma^{(n)}(u)$ is the normal stress, and V^j denote singular eigenfunctions of the elasticity operator [7]. The integration domain G is a small polygonal area around the crack tip, which can be very small and nearly arbitrary. In the context of quasi-static crack propagation, the governing equations are the equations of linear elastic theory where stresses σ and strains ε are related by Hooke’s law. The displacement field

can be calculated numerically with Galerkin finite elements. However, we are not interested in the displacement field itself but in a numerical approximation of the integral value$K(u)$. Following the approach of Rannacher and co-workers [6] within MCrack2D, we realize cell-wise error control for adaptive mesh refinement in terms of residuals and so-called dual weights. The weights z are solutions of a dual variational problem:

$$a(v, z; \Omega) = \int_{\Omega} \sigma(v) : \varepsilon(z)\, dx = K(v) \quad \text{for all} \quad v \in H^1(\Omega)$$

The main advantage of the dual-weighted-residual method is the resulting cell-based error estimator, which can be used to estimate the overall error of the functional, but also the refinement of only those parts of the mesh that really contribute to the error. The dual solution can be calculated with finite elements, but of higher order, and this makes computations very expensive. Nevertheless, the resulting adaptive strategy is more efficient than, for example, global refinement, and the main point is the precise error control. Especially in the context of crack propagation, inaccurate numerical results can have catastrophic effects, which surely justify the higher complexity of this method. Similar methods can be used for the computation of local integral characteristics. After implementing the (only briefly) described ideas for plane problems, an area for future work and still a difficult, unsolved problem is the generalization of fully three-dimensional crack growth problems.

Resource Usage

For solving the linear elasticity equations, MCrack2D uses the open-source library deal.II (www.dealii.org) based on Galerkin finite elements [2] and coupled with the p4est package (www.p4est.org). Especially p4est realizes algorithms for the distribution of triangulations over (nearly) arbitrary numbers of nodes based on MPI. Together with the PETSc library (www.mcs.anl.gov/petsc) for solving distributed linear systems coupled with the Hypre preconditioner (www.acts.nersc.gov/hypre), numerical computations in MCrack2D are truly parallel. The Arminius cluster with the InfiniBand connect is predestinated for our computations, and we use up to 10 nodes (120 cores) for computing quantities such as SIFs.

A simulation of a crack path, for example, in a CTS-specimen as shown in Figure 1 for an isotropic and a functionally graded material, needs about 50 simulation steps. In each step, the energy release rate has to be calculated, which needs at least the computation of two SIFs and three local integral characteristics for different kink angles. Again, each

computation of a SIF needs up to six - for a local integral characteristic up to twelve - mesh refinement steps, where in any refinement step two finite element solutions of different order have to be calculated.

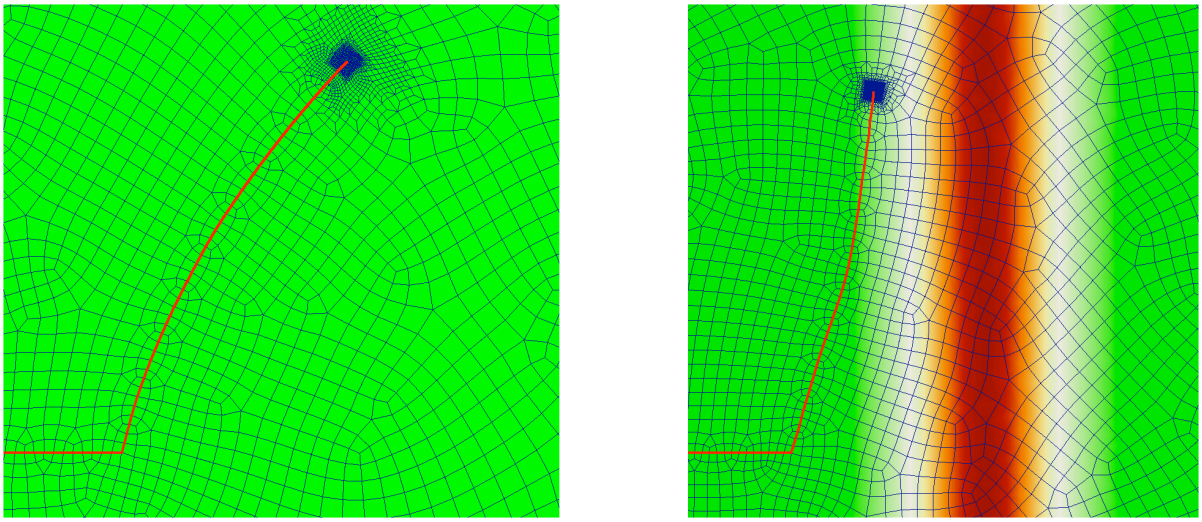


Figure 1: Simulated crack paths in a CTS-specimen, Mode-II loading, isotropic Aluminium Alloy (left), and the same scenario with a local anisotropic inhomogeneity (right)

For homogeneous materials, the local integral characteristics must be computed only once but have to be recalculated in each simulation step if the crack reaches inhomogeneous parts of the specimen. Overall, a highly accurate crack path simulation needs at least assembling and solving 50 x 2 x 6 x 2 = 1.200 linear systems with 16.000 up to 1.600.000 degrees of freedom (DoF), plus the computation of 3 local integral characteristics for up to 19 different kink angles with also at least assembling and solving 12 x 2 linear systems with 500 up to 1.890.000 DoF, plus additional computations in inhomogeneous regions of the specimen. This huge amount of numerical calculations makes it absolutely necessary to use parallel solvers.

For a demonstration, the next table shows the calculation of a SIF in a CTS-specimen with the initial and the 6-times refined mesh shown in Figure 2. Together with the estimated error, the number of DoF in the primal and dual problems plus the number of cells are shown. The grid is distributed over 120 cores, and the number of cells held on the first core is also given. In the last two columns, the total CPU time (in seconds), the sum of computing time needed by all CPUs, and the wallclock time, the real time needed from starting the simulation step to giving the final result (in seconds), is given.

Cells Core1/Total	DoF Primal	DoF Dual	Value SIF	Error	Total time (sec)	Wallclock time (sec)
66/7.933	16.618	64.978	284,05	17,01	2.210	18,5
119/14.305	30.846	121.712	285,29	7,76	2.730	22,7
223/26.833	58.462	231.796	285,96	3,81	3.660	30,5
423/50.854	110.562	439.486	286,22	1,90	5.270	43,9
805/96.670	209.166	832.994	286,38	0,96	7.880	65,7
1.533/184.015	392.764	1.565.890	286,44	0,48	12.800	107,1

In all refinement steps, the program scales nicely to the number of processors. Figure 2 also shows the distribution of the cells, for a better overview only distributed over eight cores. Within this part of the project, we computed local integral characteristics with high-accuracy for four different materials and also four complete crack path simulations. For more details see [9], [10].

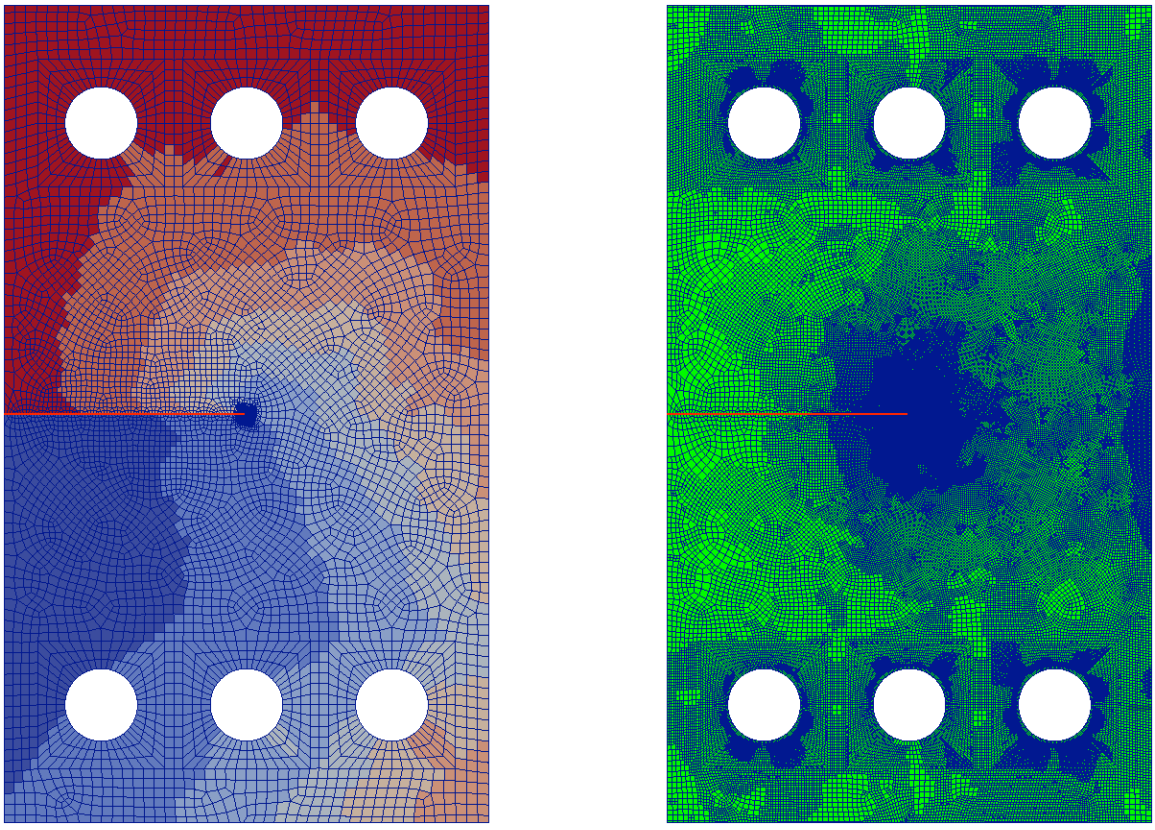


Figure 2: Initial triangulation of a CTS-specimen, distributed over eight cores (left), and the 6-times adaptively refined triangulation (right), MCrack2D

References

[1] Argatov, I. and Nazarov, S.: Energy release caused by the kinking of a crack in a plane anisotropic solid. *J. Appl. Maths. Mechs.* 2002; 66:491-503.

[2] Bangerth, W.; Hartmann, R. and Kanschat, G.: deal.II – a general-purpose object-oriented finite element library. *ACM Trans. Math. Softw.* 2007; 33(4):4.

[3] Bueckner, H.: A novel principle for the computation of stress intensity factors. *ZAMM.* 1970; 50:529-546.

[4] Griffith, A.: The phenomena of rupture and flow in solids. *Philos. Trans. Roy. Soc. London* 1921; 221:163-198.

[5] Maz'ya, V. and Plamenevsky, B.: On the coefficients in asymptotic expressions of the solutions of elliptic boundary-value problems in domains with conical points. *Math. Nachr.* 1977; 76:29-60.

[6] Rannacher, R.: Adaptive Galerkin finite element methods for partial d ifferential equations. *J. Comput. Appl. Math.* 2001; 128:205-233.

[7] Specovius-Neugebauer, M. and Steigemann, M.: Eigenfunctions of the 2-dimensional anisotropic elasticity operator and algebraic equivalent materials. *ZAMM.* 2008; 88(2):100-115.

[8] Steigemann, M.: Verallgemeinerte Eigenfunktionen und locale Integral-charakteristiken bei quasi-statischer Rissausbreitung in anisotropen Materialien. Berichte aus der Mathematik, Shaker Verlag: Aachen, 2009.

[9] Steigemann, M.: Simulation of quasi-static crack propagation in functionally graded materials. Functionally Graded Materials, Reynolds N (ed.). Nova Science Publishers, Inc. 2011.

[10] Steigemann, M.: On the precise computation of stress intensity factors and certain integral characteristics in anisotropic inhomogeneous materials. Accepted for publication in Int. J. Numer. Meth. Engng. 2012

6.3 Numerical Simulation of Fully-Filled Conveying Elements

Project coordinator	Prof. Dr.-Ing. Volker Schöppner, University of Paderborn
Project members	Dipl.-Wirt. Ing. Tobias Herken, University of Paderborn
	Dipl.-Ing. Philipp Kloke, University of Paderborn
	Dipl.-Inf. Nils Kretzschmar, University of Paderborn
Supported by	Verein zur Förderung der Kunststofftechnologie e.V.

General Problem Description

The current project aims to simulate the different processing stages of a co-rotating twin screw extruder in a fast and precise manner. That way, the efforts of costly test series can be minimized. SIGMA [1], a software that was initiated by the University of Paderborn in 1992, is able to quickly map important process values, e.g., temperature development of a thermoplastic, based on analytical equations. For a more accurate simulation, a spatially resolved consideration of the process is required. For this purpose, the Black Box module Extrud3D was developed and combined with SIGMA.

The module basically uses FeatFlow [2], a scientific code that resolves the *Navier–Stokes* equations in a robust, quick, and exact manner. The numerical calculation of screw elements requires an extremely high computing performance because each sub-process has to be considered spatially and temporally resolved. For the efficient use and further development of this system, high-performance computing (HPC) is indispensable.

Problem details and work done

SIGMA is a special simulation software that simulates different process stages of the extrusion and processing of plastics. SIGMA mainly uses one-dimensional balance equations. Therefore, it can compute the relevant process values efficiently and, above all, quickly. However from the SIGMA user's perspective, a closer look at the process values is desirable. This is, for example, the case when one tries to estimate the local overheating of the material.

For this reason, the current project aims at enabling the user to regard a sub-process spatially and temporally resolved, starting with fully-filled conveying elements. Calculations of this kind can be made with CFD (Computational Fluid Dynamics, flow simulation). IANUS Simulation GmbH will join KTP for the integration into SIGMA. IANUS is a spinoff of the TU Dortmund and works exclusively with the FeatFlow software package (developed

by the workgroup of Prof. Turek, Institute for applied Mathematics and Numerics, TU Dortmund).

FeatFlow basically solves the *Navier–Stokes equations (first and second equation; torque balance and incompressibility constraint) extended by energy conservation (third equation). The complete, fully coupled model for velocity u, pressure p, temperature Q and stress tensor t reads as follows:*

$$\rho\frac{\partial \mathbf{u}}{\partial t} + \rho(\nabla \mathbf{u})\mathbf{u} = -\nabla p + 2\eta_s \Delta \mathbf{u} + \frac{\eta_p}{\Lambda}\nabla \cdot \boldsymbol{\tau}$$

$$\nabla \cdot \mathbf{u} = 0$$

$$\frac{\partial \Theta}{\partial t} + (\nabla \Theta)\mathbf{u} = k\ \nabla^2 \Theta + \boldsymbol{\tau} : \mathbf{D}$$

The central mathematical methods for the numerical treatment of flow problems are based on LBB-stable finite element methods (Q2/P1 approach with quadratic velocity and linear pressure elements). To further reduce computing performance, the calculations were parallelized.

FeatFlow has been in scientific and industrial use for many years (via IANUS) and is considered a reliable software package. In comparison to other commercial codes (CFX, Fluent, Polyflow, etc.), the cooperation between KTP and IANUS offers the opportunity to integrate both codes, SIGMA and FeatFlow, in a way that will lead to a faster and more user-friendly package. Figure 1 is a schematic illustration of the proposed integration.

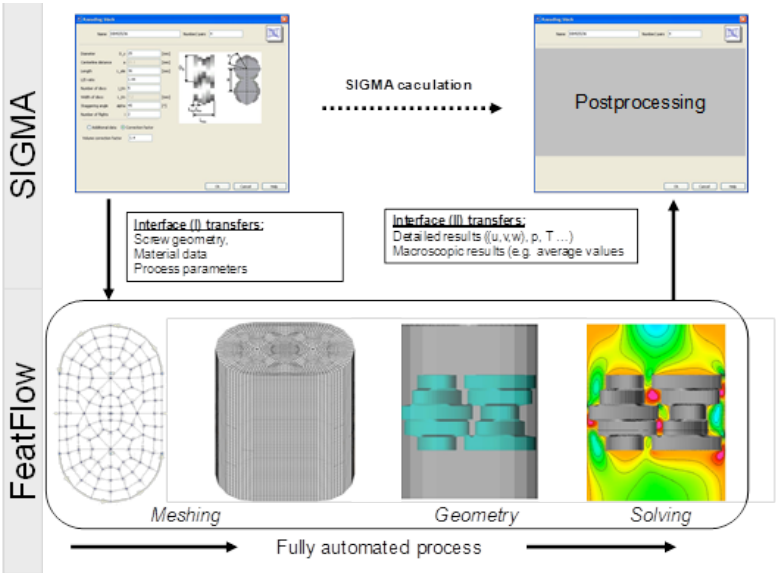


Figure 1: Schematic representation of the planned integration of Extrud3D in SIGMA

The coupling 1D-3D will gradually be realized:

1. Transfer of geometry, material data, and process parameters from SIGMA to Extrud3D
2. Extrud3D automatically processes a 2D coarse mesh and a 3D fine mesh. It also compiles a mathematical description of moving geometrical parts
3. Numerical solution of the time-dependent 3D problems
4. Transfer of the detailed results (e.g., as sectional representations of velocity, pressure, and temperature results) or in compact form

After the successful simulation, the results (see Figure 2) will be returned to SIGMA as a 1D result. The post-processing tool Paraview supports the user in examining the results in 3D.

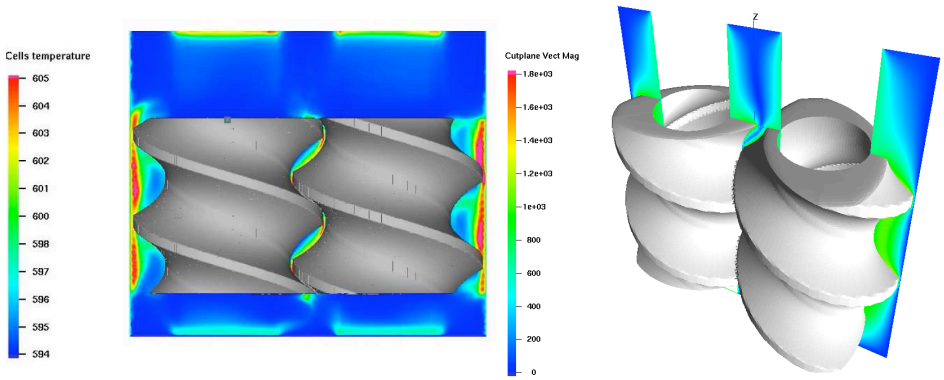


Figure 2: Numerically calculated conveying elements

Resource Usage

The FEM software package FeatFlow was developed for Linux systems and needs at least three processor cores for simulation. Therefore, its use on a conventional workstation computer, which can run SIGMA, is hard to realize. In this case, KTP utilizes the services of the PC², more specifically of the Arminius Cluster. By using 1 or 2 nodes (12 or 24 CPU cores with 36 or 72 GB Ram), the user can expect results after 14 hours of computing time. The use of the Computer Center Software (CCS) brings further benefits. It enables a more convenient resource reservation within the Arminius Cluster. The shortened computing time, achieved through parallelization, as well as the quick implementation into our system, lead to an effective further development of the software.

References

[1] Potente, H. and Thümen, A.: Method for the Optimisation of Screw Elements for Tightly Intermeshing, Co-rotating Twin Screw Extruders, International Polymer Processing (February 2006), pp. 149-154.

[2] Turek, S.: Efficient Solvers for Incompressible Flow Problems: An Algorithmic and Computational Approach

6.4 NANOHELIX

Project coordinator	Torsten Meier, Department of Physics and CeOPP, University of Paderborn
Project members	Yevgen Grynko, Department of Physics and CeOPP, University of Paderborn Jens Förstner, Department of Physics and CeOPP, University of Paderborn
Supported by:	DFG priority program SPP 1391

General Problem Description

Within the last decade, considerable progress in the field of nanooptics, which is concerned with the excitation dynamics and optical response of systems composed of different materials structured on the sub-wavelength nanoscale, has been achieved - mainly in the field of fabrication and experimental analysis. Recently, the scientific community has also begun to investigate chirality and nonlinear properties of these structures and is increasingly confronted with the lack of theories and simulation tools able to quantitatively describe the details of the combined field-material dynamics, which would allow a deeper understanding of the physical processes and the design of structures with particular desired properties.

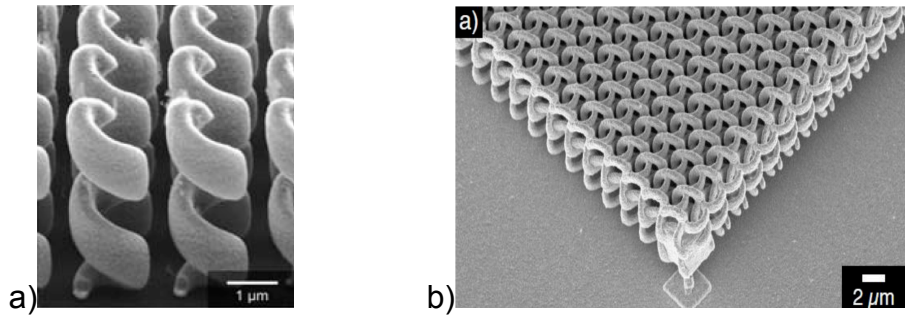


Figure 1: Array of uniaxial single helices and a bi-chiral plasmonic crystal, from [1,2].

Very recently, dielectric single-helix and bi-chiral nanostructures [1,2] (Figure 1) have been lithographically created, and it has been shown that they exhibit optical chirality, specifically that they are able to rotate the polarization of light during only a few micrometers of propagation, while natural materials like sugar solutions need several millimeters, i.e., a factor of 1000 longer distances, for the same effect. By creating metal

structures with the same geometry [3,4], these effects can even be increased over a large frequency range, using the broadband plasmonic response.
Theoretical methods and, in particular, numerical simulations can provide means for a better understanding and further prediction of the optical properties of such materials. However due to the size, shape, and variety of spatial scales, chiral and bi-chiral plasmonic crystals fabricated in the experiments appear to be a complex task for any numerical method and require significant computational power.

Problem details and work done

Since many problems in nanophotonics exhibit dynamics on different spatial scales, a multiscale Discontinous Galerkin time domain (DGTD) solver has been developed by the authors to simulate the Maxwell equations in combination with material differential equations determined by the nonlinear electron dynamics.
In this work, we apply the DGTD method [5] to simulate the experimental measurements of transmission spectra for different kinds of bi-chiral plasmonic crystals done by the group of H. Giessen at the University of Stuttgart [4].
In DGTD, the system of Maxwell equations is solved in the computational domain, discretized into a number of conforming elements of arbitrary shape. Figure 2 shows a scheme with a unit cell of a bi-chiral crystal on a glass substrate (a) and a tetrahedralized computational domain with a single helix (b). In both cases, periodic boundary conditions form an infinite structure in the *X* and *Y* directions.

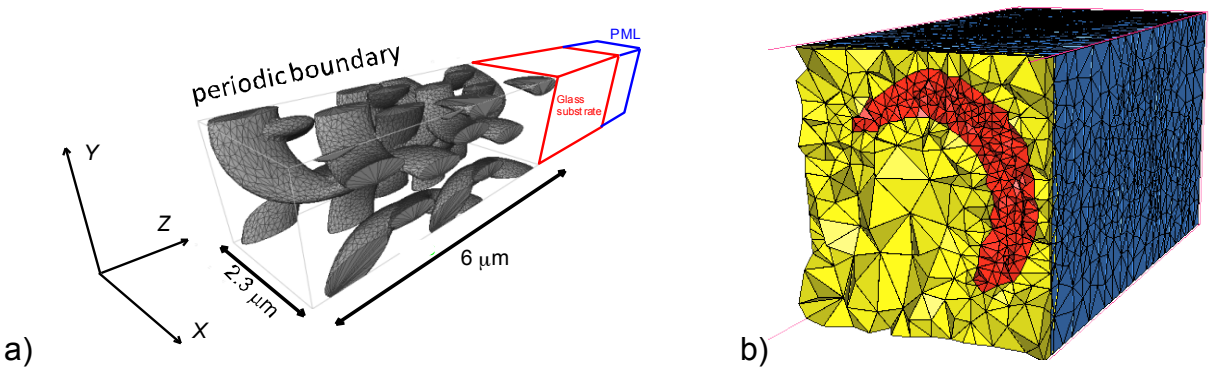


Figure 2: a) A scheme representing a computational domain with a unit cell of a bi-chiral chrystal (triangulated material interface) on a substrate. **b)** Computational mesh with a unit cell of a single-helix crystal.

A broadband pulse is used for excitation. In order to calculate the transmission of circular polarisation using linearly polarized incident light, we apply scattering matrix formalism. As a consequence in each case, two simulation runs with linear polarizations along the *X* and *Y* axes are required.

Figure 3 shows the typical property of a chiral structure: an array of single left-handed helices selectively transmits left or right circular polarization.

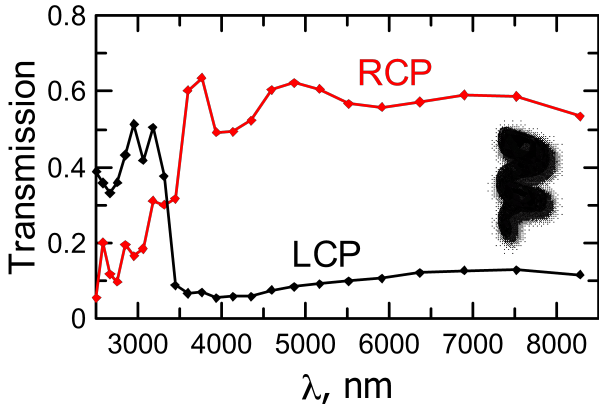
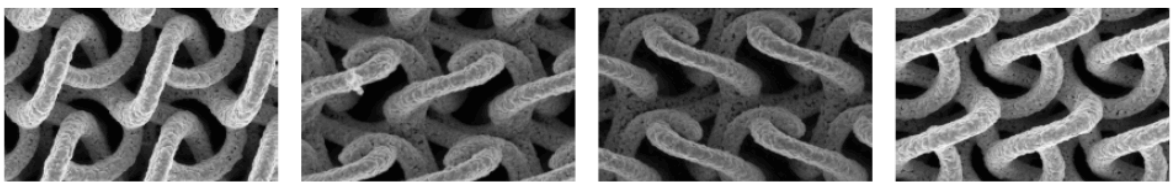


Figure 3: Simulated transmission of left circular polarization (LCP) and right circular polarization (RCP) by a left-handed single-helix plasmonic crystal.

In Figure 4, the results of the simulations for bi-chiral structures are compared to experimental measurements [4]. As bi-chiral structures possess two kinds of handedness, there are four resulting types of corresponding crystals. Our results well reproduce the phenomenon of the selective transmission of a different circular polarization. The sample with left-handed corners and left-handed helices (for the nomenclature and explanation of the experimental results, we refer to Ref. [4]) shows the RCP transmittance maximum at around 4 μm, while the LCP transmittance is twice as weak. For the right-right-handed sample, RCP and LCP curves are exchanged as it must be from the symmetry consideration.

We note that our model adequately describes the experiment and can, therefore, become a basis for theoretical studies of bi-chiral structures by means of computer simulations, which would provide further insight into their physics.

We plan to continue these simulations in the future and to study isotropic transmission properties of bi-chiral crystals, optimize geometry for efficient circular dichroism, and consider nonlinear optical properties of such structures.



Left–handed corner Left–handed helix | Left–handed corner Right–handed helix | Right–handed corner Left–handed helix | Right–handed corner Right–handed helix

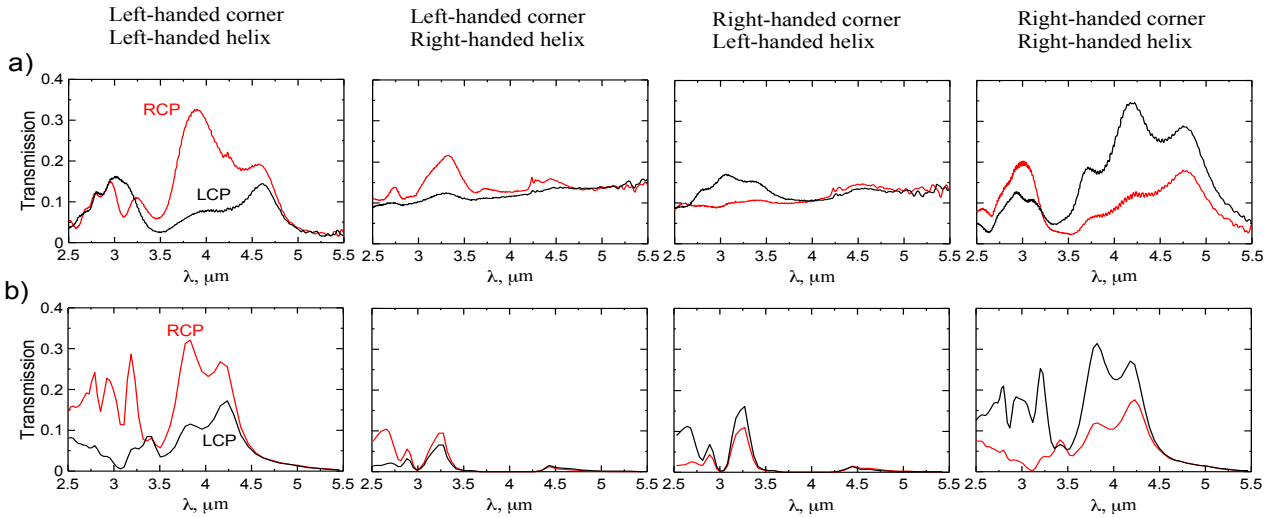


Figure 4: Measured at normal incidence (a) [4] and simulated (b) transmission for left-handed circular polarization (LCP) and right-handed circular polarization (RCP).

Resource Usage

We are using the Arminius cluster with Fujitsu RX200S6 nodes and InfiniBand HCA 4x SDR HCA PCI-e interconnections. Our code is effectively parallelized using MPI. Test calculations showed excellent scalability for at least up to 32 nodes (384 cores) (Figure 5). Therefore, we are using all available resources within the quota of our project. This makes up to 16 nodes, depending on the scale of the simulation, with weekly average frequency.

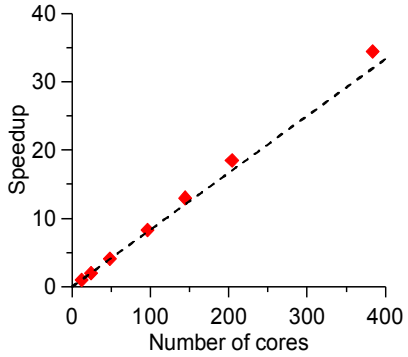


Figure 5: Parallel scalability of our code on the Arminius cluster.

References

[1] Gansel, J.K.; Thiel, M.; Rill, M.S.; Decker, M.; Bade, K.; Saile, V.; von Freymann, G.; Linden, S. and Wegener, M.: „Gold helix photonic metamaterial as broadband circular polarizer“, *Science*, 325, 1513 (2009).

[2] Thiel, M.; Decker, M.; Deubel, M.; Wegener, M.; Linden, S. and von Freymann, G.: “Polarization stop bands in chiral polymeric three-dimensional photonic crystals “, *Adv. Mater*. 19, 207 (2007).

[3] Thiel, M.; Rill, M.S.; von Freymann, G. and Wegener, M.: “Three-dimensional bi-chiral photonic crystals“, *Adv. Mater*, 21, 4680 (2009).

[4] Radke, A.; Gissibl, T., Klotzbücher, T.; V. Braun, P. and Giessen, H.: Three-Dimensional Bichiral Plasmonic Crystals Fabricated by Direct Laser Writing and Electroless Silver Plating, *Adv. Mater*. 23, 3018, (2011).

[5] Hesthaven, J. S. and Warburton, T.: Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Applications. Springer Texts in Applied Mathematics 54. Springer Verlag, New York, 2008.

[6] Grynko, Y.; Förstner, J.; Meier, T., Radke, A., Gissibl, T.; von Braun, P. and Giessen, H.: Application of the Discontinuous Galerkin Time Domain Method to the Optics of Bi-Chiral Plasmonic Crystals, TaCoNa-Photonics, 2011.

[7] Grynko, Y.; Förstner, J. and Meier, T.: Application of the Discontinuous Galerkin Time Domain Method to the optics of metallic nanostructures, AAAP Vol. 89, Suppl. No. 1, C1V89S1P041 (2011): ELS XIII Conference.

6.5 Optical control of transverse polariton patterns in semiconductor microcavities

Project coordinator	Jun.-Prof. Dr. Stefan Schumacher, Department Physik and CeOPP, University of Paderborn
Project members	Przemyslaw Lewandoswki, Department Physik and CeOPP, University of Paderborn
Supported by:	Deutsche Forschungsgemeinschaft (DFG); Project SCHU/1980-5 and research training group GRK 1461 “Micro- and Nanostructures in Optoelectronics and Photonics”

General Problem Description

Since the 70s [1], basic four-wave mixing has been known to be the origin of instability, spontaneous symmetry breaking, and related transverse pattern formation in nonlinear optical media, such as atomic vapors [2]. However, only recently it has been shown that the spatial orientation of spontaneously formed patterns can reversibly be controlled with a light beam much weaker than the actual patterns. This observation led to the proposal of a very efficient ultra-low-light-level all-optical switch with non-local action and transistor-like output performance [3] (cf. Figure 1). So far, however, the nonlinear dynamics behind the switching/redirection process are not understood. Similar optical instabilities, pattern formation, and control can be observed in quantum-well based semiconductor microcavities [4] (cf. sketch in Figure 1). In these systems, however, the nonlinearities are governed by Coulomb scattering in the semiconductor’s electronic many-particle system.

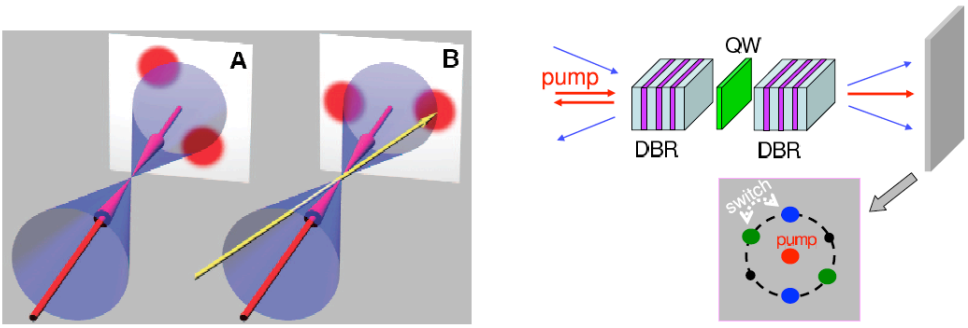


Figure 1: Visualization of the basic control scheme, where a weak control beam (yellow arrow in the left panel) can spatially redirect a much stronger pump-induced off-axis signal (big red dots on the white “screens”) in a reversible manner. The scheme is shown for an atomic vapor system (left) and a quantum-well based planar microcavity system (right). Figures taken from [4].

Problem details and work done

To study the problem outlined above, we follow two complementary routes: (i) we have derived a simple mode-competition model for anisotropic systems that can qualitatively and in a transparent manner explain the pattern formation and switching phenomena observed [5,6], and (ii) we perform fully two-dimensional, microscopically founded numerical simulations of the nonlinear excitation dynamics of the system. For the latter, we use the compute power of the PC^2 Arminius cluster.

To derive the equations our simulations are based on, we start at a microscopic fermionic Hamiltonian and capture the electronic properties of the embedded semiconductor quantum-well close to the band-gap. From here, the equations of motion for the optically induced interband polarization can be derived, which in turn couples to the optical field inside the cavity. These equations of motion can be evaluated on different levels of the theory. The simplest version, still capturing the basic nonlinearities (instantaneous excitonic scattering and phase-space filling) that lead to pattern formation phenomena as described in the introduction, is obtained in the coherent Hartree-Fock (mean-field) limit of the dynamics. We also use the effective mass approximation and assume that scattering matrix elements do not depend on the relative momentum of the scattering partners. Within this limit, an equation of motion can be derived resembling a two-dimensional Gross-Pitaevskii equation (a nonlinear Schrödinger equation) for the electronic system coupled to the coherent field, describing the dynamics of the cavity field in a single cavity mode:

$$i\hbar \begin{pmatrix} \dot{\psi}_C(x,y) \\ \dot{\psi}_X(x,y) \end{pmatrix} = \begin{pmatrix} H_C - i\gamma_1 \cdot I & g \cdot I \\ g \cdot I \cdot (1 - \alpha_1 |\psi_X(x,y)|^2) & H_X - i\gamma_2 \cdot I \end{pmatrix} \begin{pmatrix} \psi_C(x,y) \\ \psi_X(x,y) \end{pmatrix} + \begin{pmatrix} 0 \\ \alpha_2 |\psi_X(x,y)|^2 \psi_X(x,y) \end{pmatrix} + \begin{pmatrix} f(x,y,t) \\ 0 \end{pmatrix} \quad (1)$$

In addition to the usual Gross-Pitaevskii equation known, for example, from atomic condensates, loss (g_1 and g_2) and source ($f(x,y,t)$) terms are included here to allow external optical pumping, decoherence of excitonic polarization, and loss of photons from the cavity on a characteristic timescale. A more general form of this equation, including semiconductor-specific many-particle effects that go beyond the mean-field limit, will be used in future studies. In the following, we discuss results that we obtained from an explicit numerical solution of Eq.(1) in space and time. To solve this nonlinear partial differential equation, the fields y_C and y_X and operators H_C and H_X were discretized on a two-dimensional uniform grid so that Eq.(1) can be formulated as a matrix problem, which is sparse in nature. The time-evolution can then explicitly be calculated using a 4th-oder Runge-Kutta algorithm with adaptive step-size. Typical calculations are run on a single compute node with 12 cores.

In Figure 2, we show a result of our calculation, where the cavity system was excited with a continuous-wave (cw) pump in normal incidence with a flat-top Gaussian-shaped pump

intensity profile. The induced, quasi-stationary polariton density is plotted after pump excitation for about 1 nanosecond. The real-space profile reflects the pump intensity profile, whereas in the Fourier-plane (which corresponds to the expected far-field emission pattern from the cavity), the “spontaneous” formation of signals at finite **k** is clearly visible. Those signals are not included in the incoming pump field and reflect the spontaneous symmetry breaking of the nonlinear system response through polariton-wave mixing or scattering, respectively. For the isotropic system studied in this example, hexagon-formation is observed, which is consistent with experimental findings in [3].

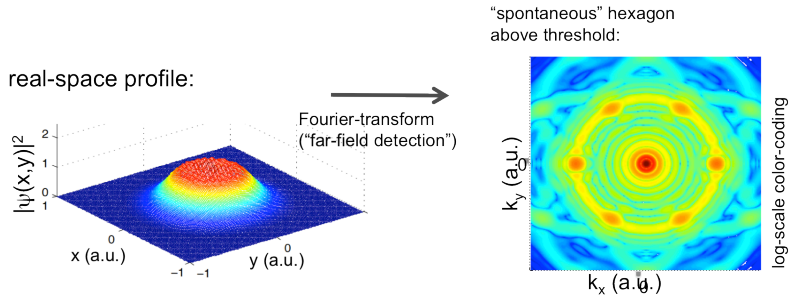


Figure 2: Left: Optically-induced stationary polariton density (setup as in Figure 1). The result is based on a solution of Eq.(1) for continuous-wave pumping in normal incidence. The Gaussian-shaped pump profile is seen in the real-space density. Right: Fourier-transform of the real-space density. The spontaneous symmetry breaking and transverse pattern formation (in this case a hexagon) is clearly visible. This k-space representation reflects the expected far-field emission from the cavity (cf. Figure 1).

In Figure 3, we demonstrate the possibility to control the spatial orientation of the transverse patters with additional external light pulses. Snapshots of the far-field emission are shown at different points in the time-evolution. At 360ns, a hexagon has spontaneously formed. An additional short light pulse centered at 600ns is then sent into the system at finite **k** (different from the six k-vectors of the bright spots on the hexagon). At 960ns, the hexagon has been reoriented by the additional light pulse and then stays stable (even after the perturbation is switched off).

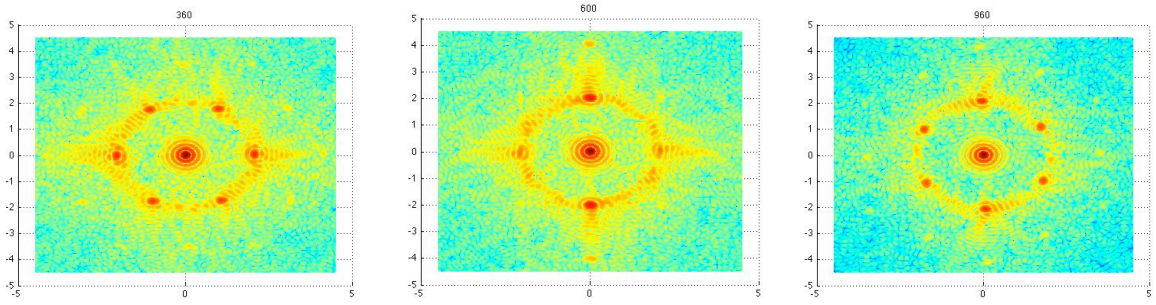


Figure 3: All-optical control of the orientation of a transverse pattern for an isotropic system (from left to right). Left (after 360ns): spontaneously formed hexagon in k-space under cw-pumping as in Figure 2. Center (after 600ns): Pattern while optically perturbing the system with an additional light pulse (control) at a finite angle of incidence (finite k-vector). Right (after 960ns): hexagon formed after the perturbation. This pattern represents the new steady-state of the system.

Future plans involve an inclusion of anisotropy, which enables us to simulate switching phenomena as discussed in the introduction. Also interesting additional effects can be expected from the inclusion of polarization dependencies and spin-dependent scattering of polaritons beyond the mean-field limit studied here [9]. This will require us to simulate the system dynamics directly in k-space, where the problem is no longer sparse in nature and the demand for computing power will increase greatly. Also, going beyond the mean-field limit will introduce quantum-memory effects in the dynamics, and consequently, the equations that need to be solved will be more complicated integro-differential equations.

Significant parts of our work have been carried out together with Rolf Binder (University of Arizona, USA) and Nai Kwong (Chinese University of Hong Kong). Some of our joint results will be presented at upcoming conferences [6,7,8].

References

[1] Yariv A. and Pepper, D.M.: *"Amplified reflection, phase conjugation, and oscillation in degenerate four-wave-mixing"*, Optics Letters **1**, 16 (1977).

[2] Grynberg, G.: *"Mirrorless four-wave-mixing oscillation in atomic vapors"*, Optics Communications **66**, 321 (1988).

[3] Dawes, A.M.C.; Illing, L.; Clark, S.M. and Gauthier, D.J.: *"All-optical switching in Rubidium Vapor"*, Science **308**, 672 (2005).

[4] Dawes, A.M.C.; Gauthier, D.J.; Schumacher, S.; Kwong, N.H.; Binder, R. and Smirl, A.L.: *"Transverse optical patterns for ultra-low-light-level all-optical switching"*, Laser & Photonics Reviews **4**, 221 (2010).

[5] Tse, Y.C.; Luk, M.H.; Kwong, N.H.; Leung, P.T.; Schumacher, S. and Binder, R.: *"A low-dimensional mode-competition model for analyzing transverse optical patterns",* in preparation.

[6] Tse, Y.C.; Luk, M.H.; Kwong, N.H.; Leung, P.T.; Schumacher, S. and Binder, R.: *"A low-dimensional population-competition model for analyzing transverse optical patterns",* to be presented at the upcoming APS March Meeting, Boston (2012).

[7] Luk, M.H.; Tse, Y.C.; Kwong, N.H.; Leung, P.T.; Schumacher, S. and Binder, R.: *"Control of transverse optical patterns in semiconductor quantum well microcavities"*, to be presented at the upcoming APS March Meeting, Boston (2012).

[8] Lewandowski, P.; Lücke, A. and Schumacher, S.: *"All-optical control of transverse polariton patterns in an anisotropic microcavity"*, to be presented at the upcoming DPG Spring Meeting, Berlin (2012).

[9] Schumacher, S.: *"Spatial anisotropy of polariton amplification in planar semiconductor microcavities induced by polarization anisotropy"*, Physical Review B **77**, 073302 (2008).

6.6 MoSGrid and related Use Cases

Project coordinator	Prof. Dr. Gregor Fels, University of Paderborn,
Project members	Dr. Brigitta Elsässer, University of Paderborn
Supported by	BMBF, D-Grid

General Problem Description

The Molecular Simulation Grid (MoSGrid) is a BMBF funded project aiming at scientists from chemistry, biology, and physics as well as related fields. It is anticipated to ease the access to high performance computing (HPC) facilities. Typically, researchers from these fields only have a limited background in modern information technology and/or computer science. Therefore, MoSGrid is aiming at reducing the initial hurdle of computational chemistry by providing tools that allow even inexperienced scientists to run molecular simulations on distributed compute infrastructures (DCI). The main focus is on an easy-to-use portal based solution enabling intuitive access. Despite basic job submission capabilities, different complex workflows for various simulation domains are also made available [1]. The convenient analysis of simulation results within one framework is only a small part. Additionally, the molecular structures can be visualized platform independent through the portal framework. Extra value is added by metadata annotation of molecular structures, simulation results, and whole workflows, enabling the retrieval of valuable data.
The project includes, among others, the Fels group and the Brinkmann group from the University of Paderborn [2, 3]. Furthermore, associated scientists such as Björn Sommer of the Hofestädt group from the University of Bielefeld have been involved with their cellmicrocosmos project.

Problem details and work done

MoSGrid requires an intuitive and stable portal framework to support functionalities, e.g., for user management, workflow handling, result visualization, and access to DCIs. A careful evaluation of available technology has shown that Unicore offers the broadest range of functions among the tested grid middleware. Additionally, it offers full workflow support including a workflow editor and resource brokerage (Figure 1). Albeit, another solution is required to cover user and data management, workflow repositories, and a robust security layer. To cover these requirements, WS-PGrade in combination with Liferay [4, 5] was applied.

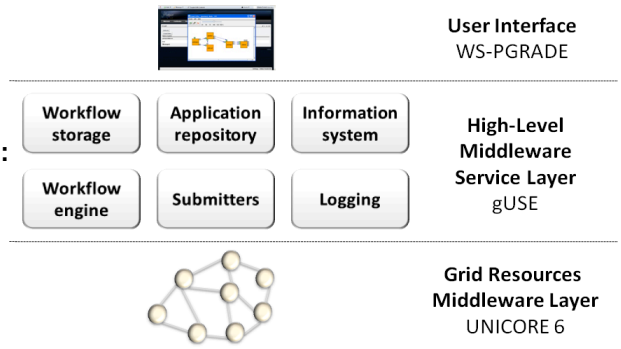


Figure 6:

The user authentication is based on grid certificates issued by registration authorities that are affiliated with the DFN / DGrid. A security portlet is used to generate a security assertion markup language (SAML) trust delegation enabling the portal server to submit jobs on the user's behalf [6, 7]. The job submission is handled by the gUSE work flow system, orchestrating single jobs [8]. Complex workflows are modeled to mimic simulation protocols used within the scientific communities (Figure 2) [9]. A customized submitter was developed to connect to the Unicore grid middleware.

The simulation data is stored on distributed data storages (XtreemFS) accessible through the portal, which is also a platform to store metadata. Results, such as energy curves or other plots, can be displayed directly from the monitoring tabs of the simulation portlets. Furthermore, molecular structures are displayed three-dimensionally using the JMol plugin. All this allows a thorough analysis at a glance, omitting the need to use various command line based applications.

Currently, three domains, archetypical for specific types of chemical simulations, are covered: (i) quantum chemical simulations, (ii) molecular dynamics, and (iii) docking. Each domain is covered by a specific portlet fulfilling the need of the particular (sub-) communities.

Practical use cases, originating from the work of involved researchers, were used to evaluate the stability and performance of the portal framework and the technologies used within. Dr. J. Krüger simulated membrane proteins and ion channels, using molecular dynamics. The simulations ranged from short minimizations to full-scale replica exchange simulations [1]. Special emphasis was put on the evaluation of simulated ion flux through channels and pores and their correlation with experimental results. Dr. M. Tusch studied the stability of amylose aggregates in various solvents by simulation. Specific molecular interactions could be identified, and their role in overall stability was demonstrated [9]. In close collaboration of B. Sommer and Dr. J. Krüger, a plugin was created enabling the direct access to DCI's via Unicore for the Membrane Editor [10, 11, 12]. The latter is an ongoing project at the University of Bielefeld, aiming at modeling and visualization of lipids and membranes. S. Rubert implemented the technology made available through the MoSGrid project during his master thesis.

Furthermore, related topics such as HPC clouds, Service-Level-Agreements, and smart application handling were studied in the context mentioned before, aiming at ease-of-usage and reliability of available technologies [3, 13].

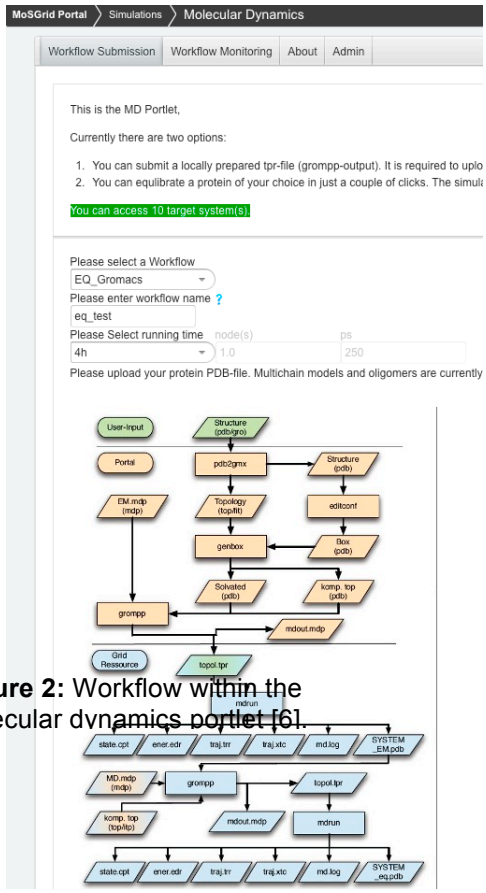


Figure 2: Workflow within the molecular dynamics portlet [6].

Resource Usage

Among the resources of the PC², mainly Bisgrid and the old and new Arminius were used. Molecular simulations have very characteristic application dependent performance profiles. Quantum chemical calculations are used to iteratively generate a partial solution of Schrödinger's equation determining the electronic structure of a molecule. They tend to be very computation and memory intense. Typically and without specific optimization, they scale up to 10-20 cores. Molecular dynamics simulations solve Newton's equation of motion yielding, the movement of atoms, and molecules under the influence of a classical force field. This type of simulation scales much further up to a couple of hundreds of cores. For specific simulation systems on suitable hardware, even superlinear scaling can be observed occasionally. Molecular dynamics simulations, involving frequent all-to-all communication, profit most from low latency networks such as Infiniband, available in aforementioned DCI's. Other simulation techniques like docking or pharmacophore search focus on the screening of large molecule databases. The individual calculations are fast and independent so that the whole simulation problem can be considered sequential.

Over the past two years, several 10.000 CPU hours have been accumulated. The volume of raw simulation data exceeds 2 TB of disk space.

The resources offered by the PC² represent an irreplaceable component in the research of all individual scientists and scientific groups mentioned before.

References

[1] Birkenheuer, G.; Breuers, S.; Brinkmann, A.; Blunk, D.; Gesing, S.; Herres-Pawlis, S.; Krüger, J.; Packschies, L. and Fels, G.: *Grid-Workflows in Molecular Science,* Proceedings of the Grid Workflow Workshop (GWW), Paderborn, Germany, February 23, 2010.

[2] Krüger, J. and Fels, G.: *Ion Permeation Simulations by Gromacs – An Example of High Performance Molecular Dynamics*, Concurrency and Computation: Practice and Experience (2010, http://dx.doi.org/10.1002/cpe.1666).

[3] Niehörster, O.; Brinkmann, A.; Fels, G.; Krüger, J. and Simon, J.: Enforcing SLAs in Scientific Clouds. In Proceedings of the 12th IEEE International Conference on Cluster Computing (Cluster2010), Heraklion, 2010.

[4] Wewior, M.; Packschies, L.; Blunk, D.; Wickeroth, D.; Warzecha, K.; Herres-Pawlis, S.; Gesing, S.; Breuers, S.; Krüger, J.; Birkenheuer, G. and Lang, U.: *The MoSGrid Gaussian portlet - Technologies for Implementation of Portlets for Molecular Simulations*, Proceedings of the International Workshop on Science Gateways (IWSG2010), ed. by R. Barbera, G. Andronico and G. La Rocca, pp. 39-43, Consorzio COMETA ISBN 978-88-95892-03-0.

[5] Gesing, S.; Marton, I.; Birkenheuer, G.; Schuller, B.; Grunzke, R.; Krüger, J.; Breuers, S.; Blunk, D.; Fels, G.; Packschies, L.; Brinkmann, A.; Kohlbacher, O. and Kozlovszky, M.: *Workflow Interoperability in a Grid Portal for Molecular Simulations*, Proceedings of the International Workshop on Science Gateways (IWSG2010), ed. by R. Barbera, G. Andronico and G. La Rocca, pp. 44-48, Consorzio COMETA ISBN 978-88-95892-03-0.

[6] Grunzke, R.; Gesing, S.; Krüger, J.; Birkenheuer, G.; Wewior, M.; Schäfer, P.; Schuller, B.; Schuster, J.; Herres-Pawlis, S.; Breuers, S., Balaskó, A.; Kozlovszky, M.; Szikszay Fabri, A.; Packschies, L.; Kacsuk, P.; Blunk, D.; Steinke, T.; Brinkmann, A.; Fels, G.; Müller-Pfefferkorn, R.; Jäkel, R. and Kohlbacher, O.: *A Single Sign-On Infrastructure for Science Gateways on a Use Case for Structural Bioinformatics*, Journal of Grid Computing (*in print*).

[8] Birkenheuer, G.; Blunk, D.; Breuers, S.; Brinkmann, A.; Fels, G.; Gesing, S.; Grunzke, R.; Herres-Pawlis, S.; Kohlbacher, O.; Krüger, J.; Lang, U.; Packschies, L.; Müller-Pfefferkorn, R.; Schäfer, P.; Schuster, J.; Steinke, T.; Warzecha, K. D. and Wewior, M.: *MoSGrid: Progress of Workflow driven Chemical Simulations*, GWW2011 (*in print*).

[7] Gesing, S.; Grunzke, R.; Balasko, A.; Birkenheuer, G.; Blunk, D.; Breuers, S.; Brinkmann, A.; Fels, G.; Herres-Pawlis, S.; Kacsuk, P.; Kozlovszky, M.; Krüger, J.; Packschies, L.; Schäfer, P.; Schuller, B.; Schuster, J.; Steinke, T.; Szikszay Fabri, A.; Wewior, M.; Müller-Pfefferkorn, R. and Kohlbacher, O.: Granular Security for a Science Gateway in Structural Bioinformatics IWSG-Life 2011 (International Workshop on Science Gateways for Life Sciences), London, UK, June 2011 (in print).

[9] Gesing, S.; Kacsuk, P.; Kozlovszky, M.; Birkenheuer, G.; Blunk, D.; Breuers, S.; Brinkmann, A.; Fels, G.; Grunzke, R.; Herres-Pawlis, S.; Krüger, J.; Packschies, L.; Müller-Pfefferkorn, R.; Schäfer, P.; Steinke, T.; Szikszay Fabri, A., Warzecha, K.; Wewior, M. and Kohlbacher, O.: A Science Gateway for Molecular Simulations In: EGI User Forum 2011, Book of Abstracts, pp. 94-95, ISBN 978-90-816927-1-7, 2011.

[9] Tusch, M.; Krüger, J. and Fels, G.: *Structural Stability of V-Amylose Helices in Water-DMSO Mixtures Analysed by Molecular Dynamics,* Journal of Chemical Theory and Computation (2011 dx.doi.org/10.1021/ct2005159, in print).

[10] Rubert, S.; Gamroth, C.; Krüger, J. and Sommer, B.: Managing GROMACS Jobs through Grid Resources using the CELLmicrocosmos 2.2 MembraneEditor IWSG-Life 2011 (International Workshop on Science Gateways for Life Sciences), London, UK, June 2011 (in print).

[11] Sommer, B.; Dingersen, T.; Gamroth, C.; Schneider, C.E.; Rubert, S.; Krüger, J. and Dietz, K. J.: *CELLmicrocosmos 2.2 MembraneEditor: A modular interactive shape-based software approach to solve heterogenous membrane packing problems*, Journal of Chemical Information and Modelling, 51(5):1165-1182, 2009.

[12] Rubert, S.; Gamroth, C.; Krüger, J. and Sommer, B.: *Grid Workflow Approach using the CELLmicrocosmos 2.2 MembraneEditor and UNICORE to commit and monitor GROMACS Jobs*, Grid Workflow Workshop GWW2011 (*in print*).

[13] Niehörster, O.; Brinkmann, A.; Keller, A.; Kleineweber, C.; Krüger, J. and Simon, J.: *Cost-Aware and SLO-Fulfilling Software as a Service*, (submitted).

6.7 Molecular Modeling studies of *Candida antarctica* lipase B catalyzed ring-opening polymerizations

Project coordinator	Prof. Dr. Gregor Fels, University of Paderborn
Project members	Iris Schönen, University of Paderborn
	Dr. Brigitta Elsässer, University of Paderborn

General Problem Description

Polyesters and polyamides are important biomaterials for medical purposes due to their biodegradability, good mechanical strength, and because they are non-toxic. Therefore, they can be employed, e.g., as surgical sutures, screws, and reinforcing plates. These polymers can be prepared by conventional chemical polymerization and alternatively by "green polymer chemistry" through lipase catalyzed polymer synthesis. In contrast to the chemical procedure *in vitro,* enzymatic catalysis is characterized by mild reaction conditions and, in addition, yields high enantio-, regio-, and chemoselectivities.

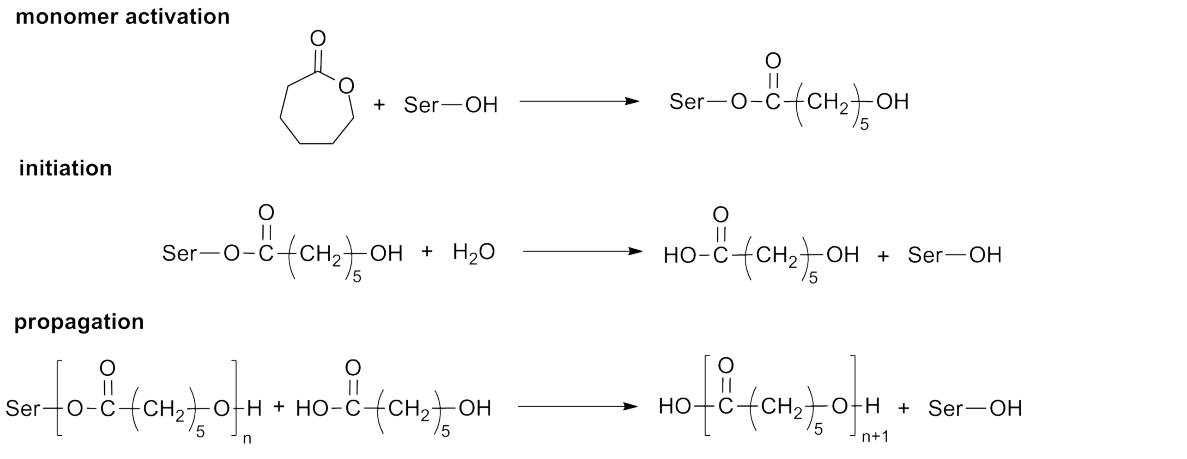
Literature provides numerous examples of enzymatic polyester formations,[1-12], which usually proceed through either a ring-opening polymerization (ROP) or a polycondensation of carboxylic acids or esters with and alcohols. In contrast, little is known about enzyme catalyzed polyamide formation.[13-15] Recently, our cooperation partner, the research group of Prof. Katja Loos at the University of Groningen, has described the first successful synthesis of a polyamide, particularly for the unbranched poly(*β*-alanine), nylon 3, by enzymatic ring-opening polymerization starting from unsubstituted *β*-lactam (2-azetidinone) using *Candida Antarctica lipase* B (CALB) immobilized on polyacrylic resin (Novozyme 435).[15] The polymerization, however, proceeds with poor yield and with a maximum chain length of only 18 units and an average length of 8.

Obviously, there is a need for a better understanding of the reaction mechanism of this enzymatic process and for enzymatically catalyzed polyamide formation in general to produce a polymer of industrial use. Theoretical studies can help to understand the underlying reaction mechanism at atomistic details and can pave the way for optimizing the enzymatic process.

Problem details and work done

The first enzyme catalyzed ring-opening polymerization was presented in 1993 by two different groups[16,17], who studied the polymerization of *ε*-caprolactone by lipase PF (*Pseudomonas fluorescens*) and porcine pancreatic lipase (PPL). The generated

polyesters have a free carboxylic acid group on one and a hydroxyl group on the other one due to the presence of water molecules in the catalytic pocket. These water molecules serve as nucleophile for the ring opening as well as for the termination of the polymerization process. The putative mechanism starts with ring opening of the lactone by the active site serine residue to yield an acyl-enzyme-complex, which then is hydrolyzed to the corresponding w-hydroxycarboxylic acid as the monomeric building block that is released into the medium. The polymerization is assumed to proceed via reaction of the ring opened monomer (rather than the lactone itself) with the acylated serine, thereby successively elongating the growing chain (Scheme 1).



Scheme 1: Putative mechanism of ROP[18]

In collaboration with the experimental work of the group of Prof. Loos, we have successfully studied the CALB catalyzed polyamide formation of *β*-lactam at atomistic details, using high level QM/MM methods. We were able to develop a reaction mechanism that is in full agreement with the experimental data.[19] The various computational methods involve, e.g., docking approaches, molecular dynamic simulations, and DFT/B3LYP QM/MM methods.

First, *b*-lactam was positioned into the active site of CALB and the enzyme-substrate complex to generate a starting structure. The protein was solvated in a 80 Å cubic box of waters and relaxed by a series of molecular dynamics steps. Then, the resulting equilibration stage of the structure was optimized using a multi-region optimization algorithm as implemented in NWChem.[20] This method performs a sequence of alternating optimization cycles of the QM and MM regions. The effective charges were recalculated in each optimization cycle by fitting the electrostatic field outside the QM region to the one produced by the full electron density representation. The complex was optimized at PBE0/DFT level of theory (with Ahlrich's pVDZ basis set), using the extensive functionality provided by the QM/MM[21,22] modules of the NWChem[20] software package. To fully

understand the catalytic mechanism at atomistic detail, QM/MM approaches combine a quantum mechanical treatment of those protein groups that are candidates for participating in the chemical reaction with a faster molecular mechanical description of the surrounding protein and solvent environment. To drive the system over the transition state to the desired intermediates and product state, harmonic restraints (springs) were imposed while, at the same time, allowing the MM system (initially equilibrated to the reactant structure) to adjust to the changes. When a reasonable estimate of the transfer was obtained, the constraints were lifted, and the system was QM/MM optimized.

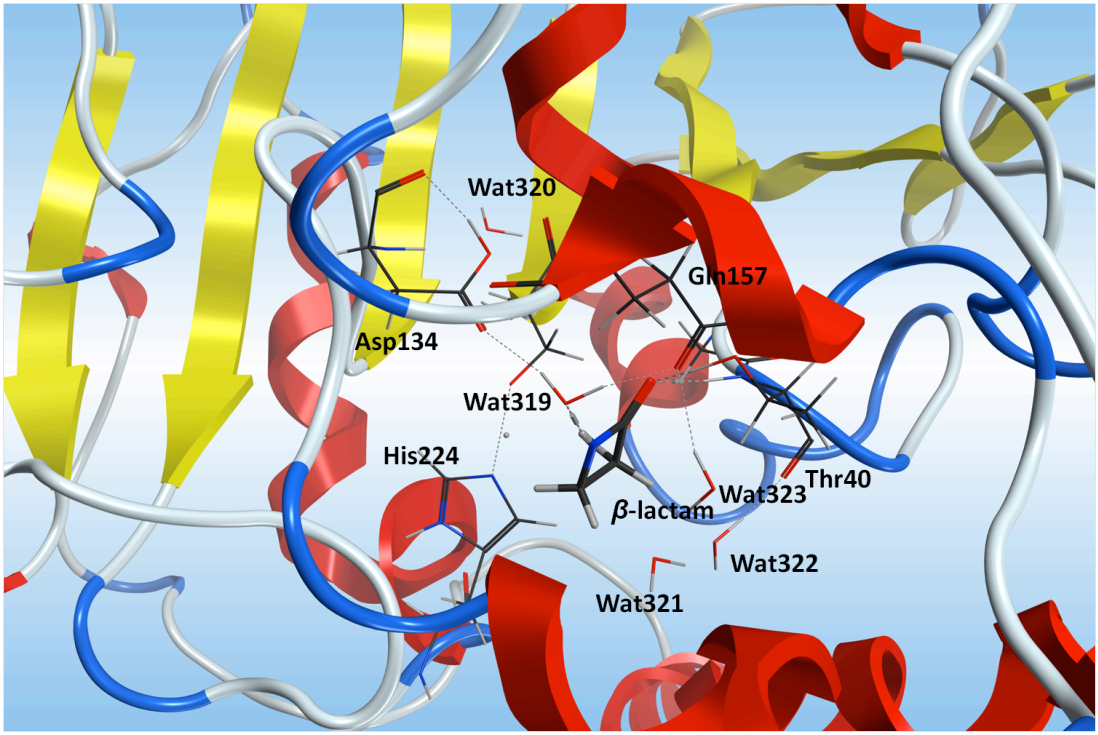


Figure 1: The active site of CALB showing *β*-lactam and the most important residues.

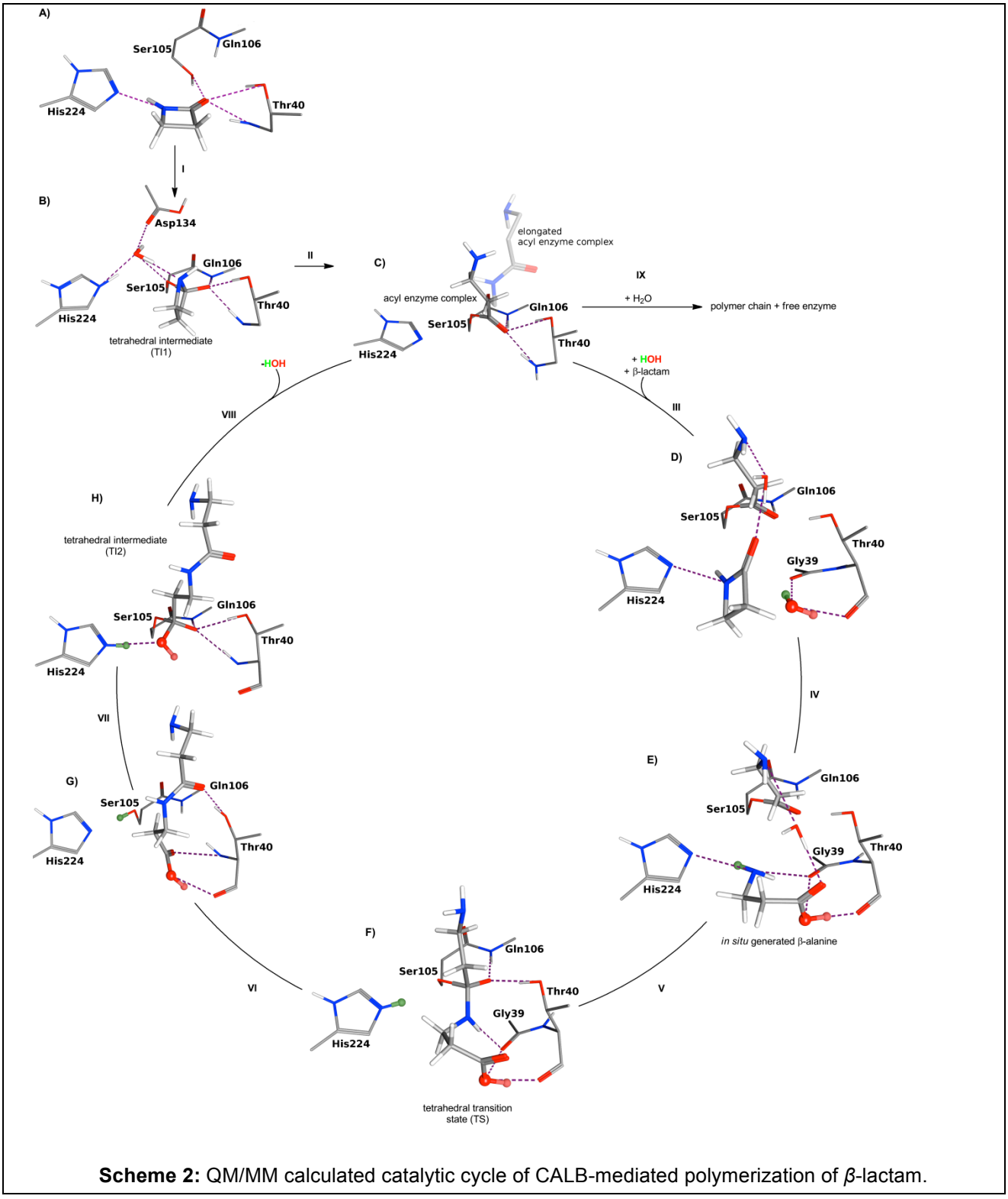
Our computational investigation of the ring-opening polymerization of *β*-lactam with CALB results in a full understanding of the necessary steps of a substrate activation, formation of the first acyl-enzyme complex, and insertion of the next lactam monomer for a chain elongation liberation of the generated dimeric amide.

As a result, we proposed a detailed catalytic cycle for the ring-opening polymerization of *b*-lactam with CALB (Scheme 2). As shown in Scheme 2, the reaction sequence was initiated by positioning of a first *β*-lactam in the active site (Figure 1, state A). In the reactant state (A), the carbonyl group *b*-lactam is coordinated by Ser105 and Thr40 and the unprotonated His224 is H-bonded to the nitrogen of the lactam ring. To simulate the attack of Ser105 on the substrate carbonyl, a spring of 1.5 Å between OG(Ser105) and the lactam carbonyl was applied, which yielded the first stabile tetrahedral intermediate (B,

TI1). The proton transfer from Ser105 to NE2(His224) took place automatically via a water molecule.

In a second set of calculations, the lactam ring was opened by transferring a proton from NE2(His224) to the lactam nitrogen, using a constraint of 0.97 Å between HE2(His224) and N(lactam). As expected, the acyl-enzyme complex (C) was formed. The process of chain elongation was simulated by first docking a next *β*-lactam to the active site, which yields a positioning of the carbonyl group towards the acyl-enzyme while the nitrogen is coordinated by His224 (step D). The activation of the second *b*-lactam took place by another active site water molecule, which generated an in situ *b*-alanine that is ready to attack the acyl-enzyme complex. This step was modeled by using a constraint between the water oxygen and the lactam carbonyl and a second spring between the lactam nitrogen and the migrating water proton. QM/MM optimization led to the in situ generated *b*-alanine (E), which was stabilized by the residues of the oxyanion hole (Thr40, Gly39).

Imposing a spring of 1.5 Å between the *b*-alanine nitrogen and the carbonyl carbon simulated the nucleophilic attack necessary for the chain elongation. Another spring was used to transfer the proton from the now positively charged nitrogen of *b*-alanine to NE2(His224). The resulting tetrahedral transition state (F) was not stable, while the QM/MM optimization of this stage resulted in the release of the dimer. Now, the dimer can either leave the active site or go on with the polymerization.



Scheme 2: QM/MM calculated catalytic cycle of CALB-mediated polymerization of *β*-lactam.

In order to continue the chain elongation, the generated dimer must translate so that the terminal carbonyl group can rebound to Ser105. This step was modeled by removing and redocking the dimer to the active site (G). For rebinding, the same spring method as

described above for stage (B) was applied and yielded the second stabile intermediate along the reaction coordinates (H, TI2). At this point, the catalytic cycle is completed and it can proceed by starting the next cycle. As an alternative, the generated polymer (oligomer) can be liberated with the aid of a water molecule. These results have recently been published in ACS Catalysis.19 It is worth mentioning that neither the lactam nor the growing chain leave the active site during elongation. A ring opening of the monomer, which yields the elongating substrate, occurs when the polymer chain is bound to Ser105.

Resource Usage

For our calculations, the highly parallel software package of NWChem20 was used installed on the Arminius cluster at the Paderborn Centre for Parallel Computing (PC^2) (60 nodes Fujitsu RX200S6, 2.67 GHz, Infiniband Switch Fabric, 7.7 TFlops Cluster with 720 cores). The resources were used daily.

References

[1] Al-Azemi, T.; Kondaventi, L. and Bisht, K.: *Macromolecules* **2002**, *35*, 3380.
[2] Binns, F.; Harffey, P.; Roberts, S. M. and Taylor, A.: *Journal of Polymer Science Part a-Polymer Chemistry* **1998**, *36*, 2069.
[3] Gross, R. A.; Kumar, A. and Kalra, B.: *Chemical Reviews* **2001**, *101*, 2097.
[4] Jääskeläinen, S.; Linko, S.; Raaska, T.; Laaksonen, L. and Linko, Y. Y. *Journal of Biotechnology* **1997**, *52*, 267.
[5] Kikuchi, H.; Uyama, H. and Kobayashi, S. *Macromolecules* **2000**, *33*, 8971.
[6] Kobayashi, S.: *Macromolecular Rapid Communications* **2009**, *30*, 237.
[7] Kobayashi, S. and Uyama, H.: *ACS Symposium series* **2003**, *840*, 128.
[8] Kumar, A. and Gross, A.: *Biomacromolecules* **2000**, *1*, 133.
[9] Thurecht, K. J.; Heise, A.; deGeus, M.; Villarroya, S.; Zhou, J. X.; Wyatt, M. F. and Howdle, S. M.: *Macromolecules* **2006**, *39*, 7967.
[10] Uyama, H. and Kobayashi, S. *Enzyme-Catalyzed Synthesis of Polymers* **2006**, *194*, 133.
[11] van der Mee, L.; Helmich, F.; de Bruijn, R.; Vekemans, J. A. J. M.; Palmans, A. R. A. and Meijer, E. W.: *Macromolecules* **2006**, *39*, 5021.
[12] Varma, I. K.; Albertsson, A.-C.; Rajkhowa, R. and Srivastava, R. K.: *Prog. Polym. Sci.* **2005**, *30*, 949.
[13] Cheng, H. N. and Maslanka, W. W.; Gu, Q.-M. *[US 6677427* **2004**, *Hercules Inc., invs.*

[14] Gu, Q.-M.; Maslanka, W. W. and Cheng, H. N. *Polymer Biocatalysis and Biomaterials II, Editor(s): H. N. Cheng, R. A. Gross, ACS Symposium series* **2008**, *999*, Chapter 21.
[15] Schwab, L. W.; Kroon, R.; Schouten, A. J. and Loos, K.: *Macromol. Rapid Commun.* **2008**, *29*, 794.
[16] Knani, D.; Gutman, A. L. and Kohn, D. H. *J Polym Sci Pol Chem* **1993**, *31*, 1221.
[17] Uyama, H. and Kobayashi, S. *Chem Lett* **1993**, 1149.
[18] Kobayashi, S.; Uyama, H. and Kimura, S. *Chem Rev* **2001**, *101*, 3793.
[19] Baum, I.; Elsasser, B.; Schwab, L. W.; Loos, K. and Fels, G. *Acs Catal* **2011**, *1*, 323.
[20] Valiev, M.; Bylaska, E. J.; Govind, N.; Kowalski, K.; Straatsma, T. P.; Van Dam, H. J. J.; Wang, D.; Nieplocha, J.; Apra, E.; Windus, T. L. and de Jong, W. A.: *Computer Physics Communications* **2010**, *181*, 1477.
[21] Valiev, M.; Garrett, B. C.; Tsai, M. K.; Kowalski, K.; Kathmann, S. M.; Schenter, G. K. and Dupuis, M.: *J Chem Phys* **2007**, *127*, 51102.
[22] Valiev, M.; Kawai, R.; Adams, J. A. and Weare, J. H.: *J Am Chem Soc* **2003**, *125*, 9926.

6.8 The Role of Protonation in the Ribonuclease A Transphosphorylation Reaction

Project coordinator	Prof. Dr. Gregor Fels, University of Paderborn
Project members	Dr. Brigitta Elsässer, Universtiy of Paderborn
Supported by:	University of Paderborn

General Problem Description

Bovine pancreatic Ribonuclease A (RNase A) catalyzes the cleavage of single stranded RNA in two steps.[1] In the first transphosphorylation step, a nucleophilic attack of a ribose O2´ on the scissile phosphate (3´,5´-phosphodiester) yields a cyclic phosphate and a 5´OH-product. In a second consecutive step, this product is then hydrolyzed to the 3´-phosphate after a water attack on the phosphorus (Figure 1). We have previously studied the hydrolysis step of the reaction[2] and now turn our attention to the transphosphorylation mechanism.

Bovine pancreatic Ribonuclease A (RNase A) accelerates the cleavage of single stranded RNA with rates 10^{12} times faster than the spontaneous uncatalyzed reaction.[3] Although it has been studied since the 1950s, there is still no complete understanding of how this enzyme achieves this remarkable reaction rate up, and the details of its mechanism are widely debated. In the present theoretical study, we analyze two limiting reactions pathways and free energy profiles, using DFT based QM/MM methodology.[4,5] During our calculation, the large (117 atoms) active site was accurately treated quantum chemically in the QM region, and the surrounding protein with the solvent was described by well established force field methods at classical MM level.

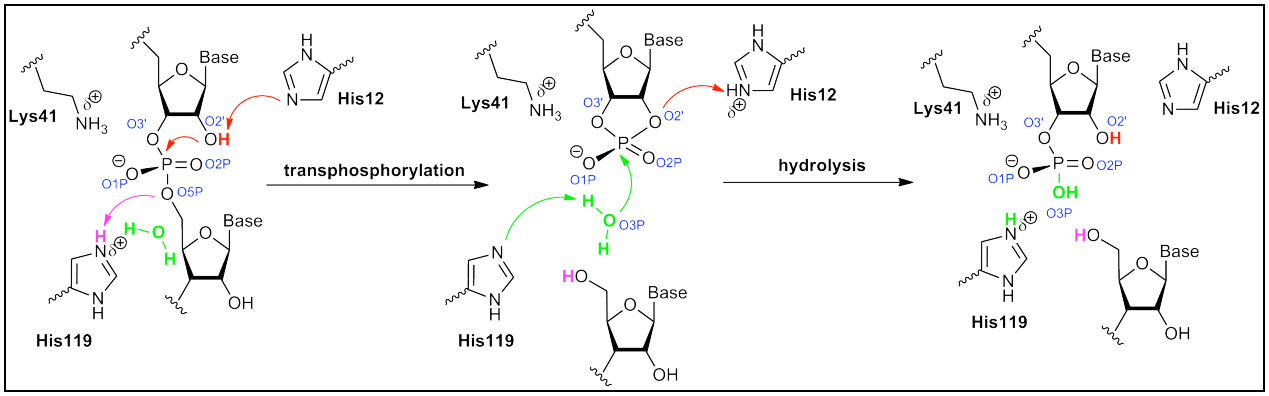


Figure 1: Putative mechanism of the enzymatic phosphodiester cleavage of RNase A[1]

Problem details and work done

The most important question addresses the nature of the phosphorane intermediate regarding the stability of the phosphorane and its protonation state. The concepts of the proposed mechanism are not new[6-10] but are not commonly accepted and are based on either experimental observation or theoretical studies of small fragments of the active site residues. According to the general acid-general base mechanism[1] (path1, Figure 2) in the first step, the proton is transferred from O2´ to the general base His12, followed by a nucleophilic attack of O2´ on the scissile phosphorous. The generated dianion is considered to be unstable. Therefore in the next concerted step, a proton is transferred from His119 (general acid) to O5P, and the P–O5P bond is being cleaved. The modified Breslow mechanism[7] implies a two step proton shuttle, first to the O2P and then to the O1P oxygen of the phosphorane (triester mechanism). These steps yield a monoanionic pentacoordinated intermediate.

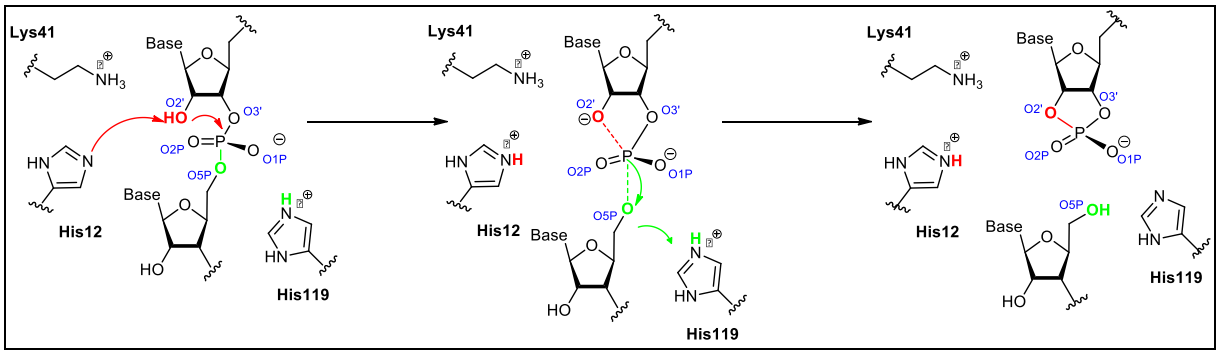


Figure 2: Putative reaction mechanism (path 1) of the RNase A transphosphorylation with direct proton transfer.[1]

In the last step, the proton from O1P is transferred to His12 and from His119 to O5P to complete the cleavage of the leaving group. There are pros and cons to both suggested mechanisms. Gas phase calculations suggest that the transition state must be neutralized (proton added) or the charge build up will be too large, leading to a very high activation barrier and no reaction. However, it has been shown that a Born-type model of solvation can stabilize a dianionic phosphorane.

During our computational study, we compared the monoanionic to the dianionic path and found additional evidence that the proton shuttle from O2´ to O1P directly yields a stable monoanionic intermediate (path 2, Figure 3). In addition, it lowers the reaction barrier enormously. The first path, which proceeds over a dianionic phosphorane, has no stable intermediate stage, and the monophosphate transition state is more likely to be dissociative. In contrast, the second path is a two step procedure with a stable pentacoordinated associative monoanionic intermediate and a much lower reaction barrier (Figure 3).

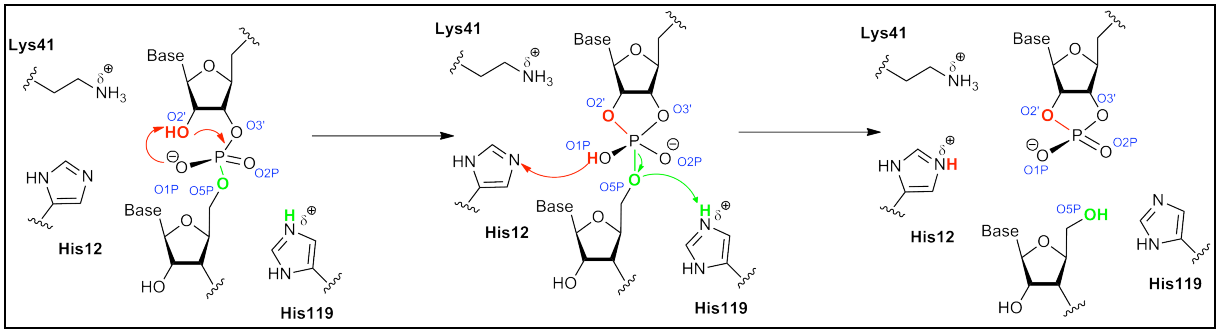


Figure 3: Calculated lower barrier reaction mechanism (path 2) for the RNase A transphosphorylation step.

Contrary to prior theoretical investigations[6-10], we could model the whole enzyme-ligand-solvent complex, using the QM/MM module of NWChem. Theoretical calculations require atomistic details on starting geometry of the molecules involved, which can only be achieved through computational docking studies in the absence of crystallographic data. Fortunately, docking algorithms have been improved tremendously during the past years so that today reliable structures of enzyme-ligand complexes can successfully be obtained from computation, applying the well-known protein ligand interactions of the reactant state as described in literature.[1] In addition, we could also compare our docking results with the desoxy-cytidyl-adenosine-RNase A complex, which is available in the PDB database (1RPG).[11] Since RNase A only cleaves the single stranded RNA behind uridyl and cytosyl nucleotides, uridyl nucleotide oligomers were placed into the active site of RNase A in the presence of crystallographic water molecules for the docking simulation.[12]

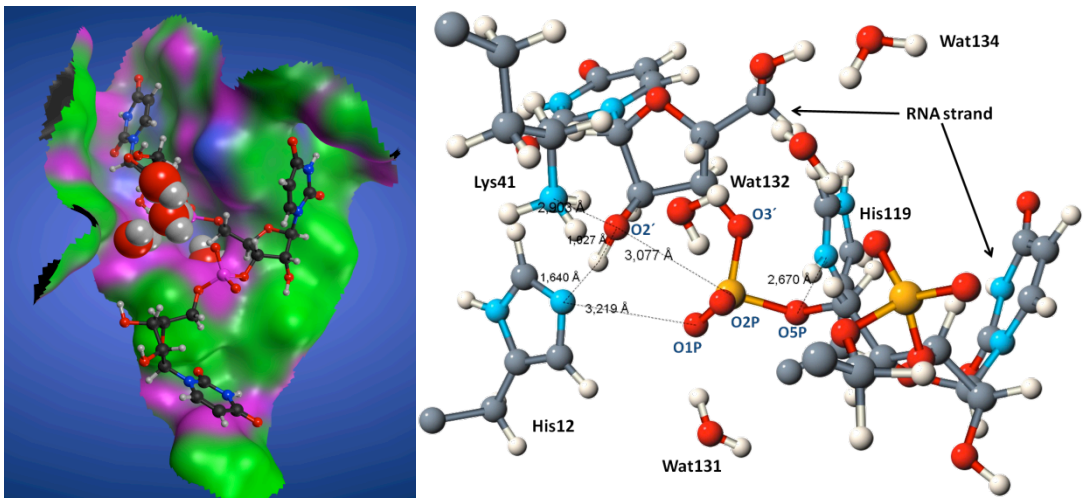


Figure 4: QM/MM optimized docked reactant state of the RNase A transphosphorylation step. (left) The displayed surface illustrates the catalytic pocket. Waters are represented by space filling. RNA strand is displayed as balls and sticks. (right) The QM region of the reactant state shows the most relevant H-bond distances.

The resulting hits were analyzed with respect to the length of the following important H-bond distances (NZ)Lys41–O2´, (NE2)His12–O2´, (NE2)Glu11–O2P, (NE2)His12–O1P, and (ND1)His119–O5P. The results of the docking simulations are published elsewhere[12]. Finally, the top docking hit was used as an initial structure input for our calculations (Figure 4).

The system was divided into a region to be treated with quantum mechanical methods, the QM region, and the remaining protein, counter ions, and solvent molecules, which were described by using a molecular mechanics model, the MM region. The forces in the QM subsystem were calculated at the PBE0[13] level, using Ahlrichs-pVDZ basis set[14]. The MM region was described using the AMBER99 force field. The bonds between the QM and MM subsystems were capped with H-atoms[4]. First, the entire solvent-enzyme-ligand structure was equilibrated by performing a series of molecular dynamics annealing runs for 100 ps at temperatures 50 K, 150 K, 200 K, 250 K, and 298.15 K with fixed positions of the atoms in the QM region. The positions in the MM region were then fixed, and the atomic positions in the QM region were optimized at the PBE0 level. Several cycles of this optimization were carried out until convergence was obtained. To allow the system to reorganize and to avoid being trapped in a metastable minimum, we equilibrated the enzyme-substrate complex prior to the QM/MM optimization of the entire system.

In the reactant state, the P–O2´ bond length is 3.07 Å, the (NZ)Lys41–O2´ and (NE2)His12–O2´ are 2.90 and 2.64 Å, respectively. Furthermore, O1P is weakly H-bonded to (NE2)His12 (3.22 Å), and the (ND1)His119–O5P distance is as short as 2.67 Å (Figure 4). Based on this initial structure, we generated reaction pathways towards a/the product state by decreasing the P–O2´bond length (path1) and lengthening the O2´–H(O2´) bond (path2), respectively.

In path 1, a spring of 1.7 Å was applied to the P–O2’ bond, and the constrained optimization resulted in an unstable dissociative TS as presented on Figure 4. After removal of the spring, the structure relaxed to the desired product state. This reaction passes through a dianionic transition state (TS) with a barrier height of 25.09 kcal/mol and reaches the product stage without the formation of a stable intermediate state (Figure 5, left). In the transition state, the proton from O2´ has already been transferred to His12, and the negatively charged O2´ is now ready to attack the phosphorous. However, the P–O2´bond is not formed yet and has a bond order[15] of almost 0. At the same time, His119 has started to shuttle its proton to O5P, and the P–O5P bond is already broken. Therefore, this transition state is rather dissociative, and the phosphate is more likely to be a metastable monophosphate, where the negative charge is shared between O1P and O2P (Figure 5, right). As the reaction proceeds, the P–O2´bond forms to generate the cyclic phosphate as a product of the transphosphorylation step with a total free energy change of -7.32kcal/mol.

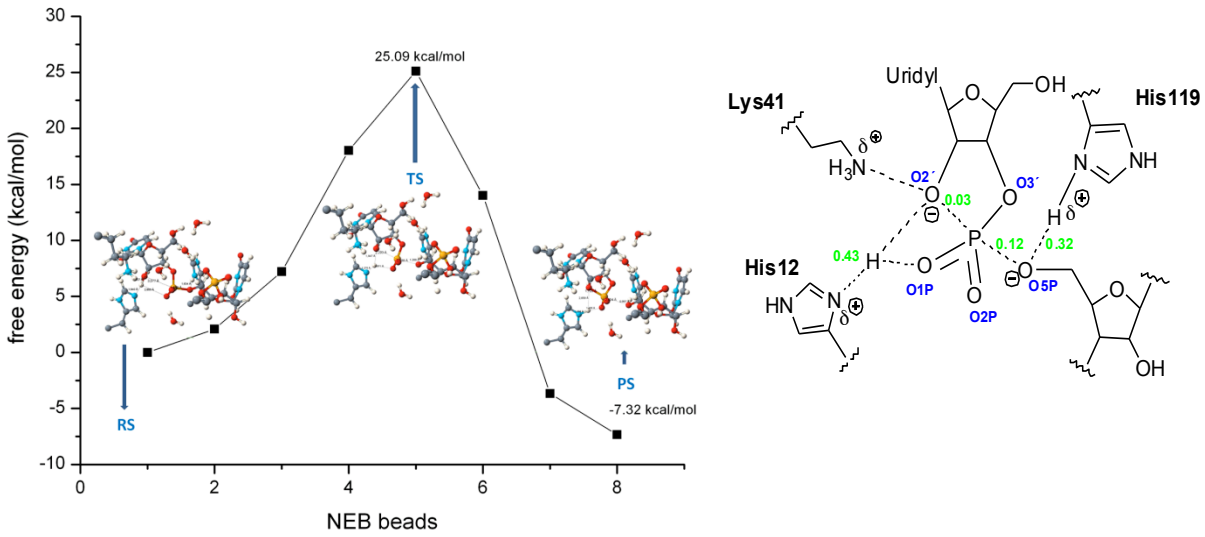


Figure 5: (left) Free energy profile of path1. (right) Transition state structure. The green numbers designate bond orders.[15]

In path2, the O2´–H(O2´) bond is elongated and optimization results in a proton transfer to O1P and the formation of a stable monoanionic pentacoordinated phosphorane intermediate structure. Since, the charge on the phosphate has been neutralized, the barrier from reactant to intermediate is 9.84 kcal/mol (TS1), and is, therefore, much lower than the activation energy of path1 (Figure 6, left). The reaction begins slowly by stretching out the O2´–H(O2´) bond, and the proton moves first towards His12. At bead4, the TS1 the proton is shared between His12 and O2´. However, the O2´–H and H–(NE)His12 distances are relatively long with 1.53 Å. Afterwards, the proton moves closer to O1P, and at bead5 the migrating proton is shared between O2´, O1P, and His12. Then, it continues to be shifted toward O1P. At TS1 (Figure 7, bead6), the proton starts getting transferred (O1P–H is 1.46 Å) and is still weakly coordinated by His12 (His12–H is 1.51 Å). At the same time, the proton is already completely departed from O2´ (O2´–H is 1.94 Å). As the reaction carries on to INT, the now negatively charged O2´ attacks the phosphate.

In the intermediate state (INT, bead 8), the newly formed P–O2´ bond has a calculated bond order of 0.82, and the P–O5P bond has not been cleaved yet. The proton of O2´has been fully transferred to O1P and is strongly H-bonded to His12 (2.61 Å) (Figure 6, right).

To reach the desired product state, the P–O5P was elongated, and the reaction proceeds first slowly over a second, but much smaller, barrier with a height of 6.79 kcal/mol (TS2, bead11). To reach TS2, the P–O5P gets longer and both the O1P and the proton of His119 start migrating towards His12 and O5P, respectively. At TS2, these protons are shared between the reaction partners (the average H–O and H–N bond length is 1.35 Å) and the expanded P–O5P bond shows a bond order of only 0.22. Subsequently, the P–

O5P bond breaks completely, and the proton from O1P is fully transferred to His12 and from His119 to O5P.

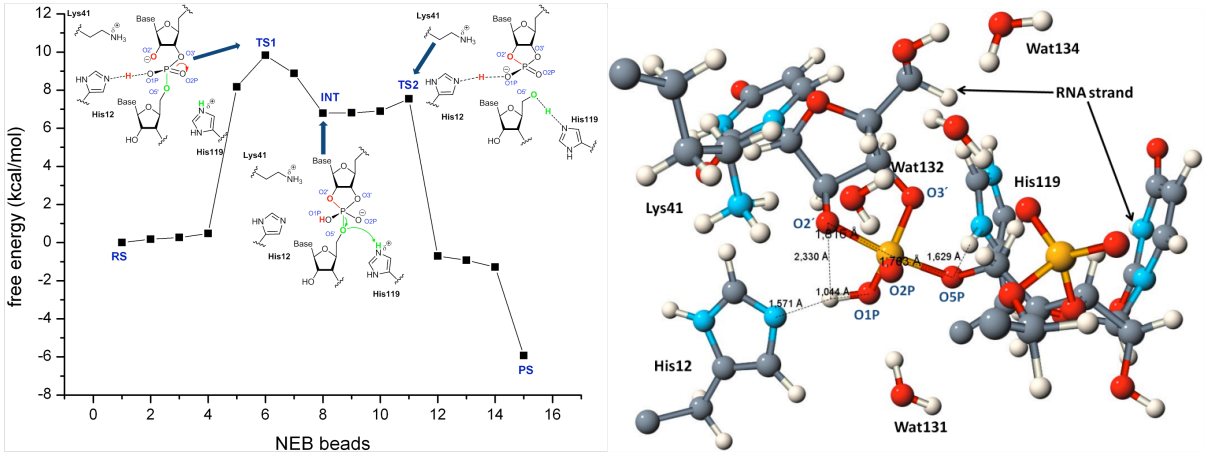


Figure 6: (left) Free energy profile of path2. (right) Stable pentacoordinated phosphorane intermediate state (INT).

Since the O5´-nucleotide is being cleaved at this stage of the RNase A hydrolysis, we suppose that it leaves the active site before the reaction proceeds with the hydrolysis of the cyclic phosphate. The product states of path1 and path2 were compared and are very close in energy (difference: ~1.4 kcal/mol) as well as in structure.

The detailed reaction paths from reactant to intermediate and from intermediate to product state were obtained using a QM/MM nudged elastic band (NEB) methodology.[5,16] In the NEB calculations for this work, 8 beads were used for path1 and a total of 2x8 beads/replicas for path2 in separate calculations towards product and/or intermediate state starting from the reactant structure. The initial guess for the pathway was generated by linear interpolation between optimized reactant and intermediate and product states. The system was equilibrated at each NEB node along the reaction path, allowing the remaining protein and the solvent to respond to the movement along the reaction coordinate. To properly account for finite temperature fluctuations of the protein, Helmholtz free energy along the reaction path was calculated, providing accurate estimations of activation barriers[5].

Additional support for this mechanism is gained from the fact that the calculated product state of the transphosphorylation step is entirely consistent with the found reaction state structure of the hydrolysis step published earlier. The calculated low barrier of path2 also supports the conjectures of Westheim[17], Breslow[6], and Warshel[7] and complies with the results of the kinetic isotope studies of Harris et al.[18]

Due to the high level of these calculations, which we believe include accurate estimates of the essential interactions in the system as well as thermal averaging (free energy estimates), we believe that the results presented here provide strong support for the role of

multi-step proton transfer in the RNase A transphosphorylation step and emphasize the presence of a stable phosphorane intermediate in this important catalytic mechanism. The additional elements included in this calculation make us to be the first group who succeeds in interpreting the complete enzyme mechanism at such high level of calculation.

Resource Usage

For our calculations, the highly parallel software package of NWChem[19] was used, installed on the Arminius cluster at the Paderborn Centre for parallel computing (PC^2) (60 nodes Fujitsu RX200S6, 2.67 GHz, Infiniband Switch Fabric, 7.7 TFlops Cluster with 720 cores). The resources were used daily.

References

[1] Raines, R. T. Chemical Reviews 1998, 98, 1045-1065.
[2] Elsaesser, B.; Valiev, M. and Weare, J. H. J. Am. Chem. Soc., 2009, 131, 3869-3871.
[3] Emilsson, G. M.; Nakamura, S.; Roth, A. and Breaker, R. R. RNA-A Publication of the RNA Society 2003, 9, 907-918.
[4] Valiev, M.; Garrett, B. C.; Tsai, M. K.; Kowalski, K.; Kathmann, S. M.; Schenter, G. K. and Dupuis, M. Journal of Chemical Physics 2007, 127, 51102.
[5] Valiev, M.; Yang, J.; Adams, J. A.; Taylor, S. S. and Weare, J. H. Journal of Physical Chemistry B 2007, 111, 13455-13464.
[6] Breslow, R.; Dong, S. D.; Webb, Y. and Xu, R. J. Am. Chem. Soc., 1996, 118, 6588-6600.
[7] Glennon, T. M. and Warshel, A. J. Am. Chem. Soc., 1998, 120, 10234-10247.
[8] Lim, C. and Tole, P. J. Am. Chem. Soc., 1992, 114, 7245-7252.
[9] Perreault, D .and Anslyn, E. Angew. Chemie International Edition 1997, 36, 432-450.
[10] Wladkowski, B. D.; Krauss, M. and Stevens, W. J. J. Am. Chem. Soc., 1995, 117, 10537-10545.
[11] Zegers, I.; Maes, D.; Daothi, M. H.; Poortmans, F.; Palmer, R. and Wyns, L. Protein Science 1994, 3, 2322-2339.
[12] Elsaesser, B. and Fels, G. J. Mol. Mod. 2011, 17, 1953-1962.
[13] Adamo, C. and Barone, V. Journal of Chemical Physics 1999, 110, 6158-6170.
[14] Schafer, A.; Horn, H. and Ahlrichs, R. J. Chem. Phys. 1992, 97, 2571.
[15] Berente, I.; Beke, T. and Naray-Szabo, G. Theoretical Chemistry Accounts 2007, 118, 129-134.
[16] Henkelman, G. and Jonsson, H. Journal of Chemical Physics 2000, 113, 9978-9985.
[17] Westheim, F. Accounts of Chemical Research 1968, 1, 70-&.

[18] Harris, M. E.; Dai, Q.; Gu, H.; Kellerman, D. L.; Piccirilli, J. A. and Anderson, V. E. J. Am. Chem. Soc., 2010, 132, 11613-11621.

[19] Valiev, M.; Bylaska, E. J.; Govind, N.; Kowalski, K.; Straatsma, T. P.; Van Dam, H. J. J.; Wang, D.; Nieplocha, J.; Apra, E.; Windus, T. L. and de Jong, W. A. Computer Physics Communications 2010, 181, 1477-1489.

6.9 Adsorption of organic adhesion promoters on magnesium oxide surfaces

Project coordinator	Prof. Dr.-Ing. Guido Grundmeier, University of Paderborn
Project members	Dr.-Ing. Ozlem Ozcan, University of Paderborn

General Problem Description

The use of light metals in automotive and aircraft industries offers great potential to reduce the vehicle weight and already reduces the fuel consumption. Magnesium alloys like AZ31 have a 30 % lower density than aluminum alloys. However, such alloys have a lower corrosion resistance, especially in chlorine containing electrolytes. A possible alternative to the standard chromate containing treatments, which have to be replaced because of their harmful environmental impact, is the application of self-assembled monolayers (SAMs) [1,2]. A densely packed monomolecular layer can prevent a direct contact between corrosive ions and the metal surface [3]. Moreover through the selection of appropriate functional head groups, additional linking of organic coatings to bi-functional monolayers can be achieved resulting in an improved adhesion [4]. The success of such systems was already demonstrated for aluminum alloys in terms of corrosion protection and adhesion promotion [5,6].

The observed long-term effect in corrosion protection and adhesion promotion depends on the strength of the bonds at the SAM – metal oxide interface in presence of water. Therefore, it is crucial to understand the binding mechanism of those molecules to the metal oxide surface as well as to evaluate their binding energies in presence of different water structures to reach conclusions that can guide us in the interpretation of our experimental observations.

The project involves polar and non-polar single crystalline MgO surfaces to enable a comparison between the various possible crystalline orientations, approaching the chemistry of the alloy surfaces. The molecules used in this study are phosphonic acid molecules with different alkyl chain lengths. The analysis aims at understanding the binding mechanisms of those molecules to MgO surfaces as well as assessing the stability of those bonds against water.

Problem details and work done

The SIESTA [8] code was used to study the adsorption of organic phosphonic acid molecules on MgO (001) and MgO (111) surfaces. SIESTA is a simulation package that allows the handling of relatively large system sizes, which is necessary for the

investigation of stepped surfaces as well as the evaluation of binding energies of isolated molecules. For both crystal surfaces, 2 x 2 x 3 size slabs were used for the calculation, and 25 Angstroms of vacuum were placed between the slabs to prevent undesired interaction. For all atoms, DZP (double zeta polarized) basis sets were used.

Calculations were performed using density functional theory with the generalized-gradient approximation (GGA) and utilizing the exchange-correlation potential developed by Perdew, Burke, and Ernzerhof (PBE) [7]. The SIESTA [8] code was employed with its localized atomic orbital basis sets and pseudopotential representation of the core states. The techniques used to derive pseudopotentials for this work were based to a great extent on the approach described by Giannozzi et al. [9]. The SIESTA pseudopotentials (PP) are generated using the program ATOM supplied as part of the SIESTA program package. For the PP generation, the PBE functional was used to comply with the main SIESTA calculations [7,8]. Relativistic Troullier–Martins pseudopotentials [10] with non-linear core corrections [11] were implemented in their fully non-local form [12-13].

In literature, quantum chemical calculations [14,15] mostly agree with the prediction that water does not dissociate on the perfect MgO (100) surface. Our initial results were in good agreement with the literature, and no dissociation of water was observed. On the MgO (001) surface, the binding energy of the probe molecule MPA (methyl phosphonic acid) was calculated for different surface coverages on clean surfaces and surfaces containing a molecularly adsorbed water layer. On clean surfaces, the bi-dentate binding structure resulted in the highest adsorption energies, whereas on molecularly adsorbed water layers, the computations indicated a hydrogen-bonded state.

Coverage dependency of binding energies was not observed. This may be explained by the short chain length (one methyl group) of the chosen probe molecule, also not leading to a self assembly in experimental studies. Currently, longer chains of alkylphosphonic acids with odd and even numbers of methylene groups are being investigated in terms of their coverage dependent adsorption energies on MgO (001) surfaces.

The MgO is a strongly bonded ionic oxide with alternating atomic planes of cationic (Mg^{2+}) and anionic (O^{2-}) in the fcc ABC-type stacking along the [111] direction. Therefore, the generated surfaces of MgO (111) carry a dipole moment, which can be stabilized at low temperatures by adsorption of hydroxyls [16]. Our calculations have shown that the MgO (111) surface is stabilised by a full coverage of hydroxyls. Initial calculations of MPA adsorption on those surfaces also indicated a hydrogen bonded state.

A paper focusing on the comparison of the adsorption and stability of organic phosphonic acids on MgO (001) and MgO (111) single crystalline surfaces, combining our experimental and computational results is in preparation and expected to be ready for submission in April 2012.

Resource Usage

SIESTA code is a very suitable code for the calculation of large systems since the computation time scales almost linearly with the system size and the parallelization is achieved efficiently. This is very important for our surface calculations since the realistic simulation of surface defects, reconstructions, and adsorbed species on these surfaces requires large cell sizes and, thus, a high number of atoms to be considered. Therefore, the computations were performed on the ARMINIUS cluster. Calculations started in April of 2010, stalled during the maintenance, and resumed in 2011.

References

[1] Ulman, A.: Chem. Rev. 96 (1996) 1533.
[2] Lim, M.S.; Feng, K.; Chen, X.; Wu, N.; Raman, A.; Nightingale, J.; Gawalt, E.S.; Hornak, L.A. and Timperman, A.T.: Langmuir 23 (2007) 2444.
[3] Thissen, P.; Janke, S.; Feil, F.; Fürbeth, W., Tabatabai, D. and Grundmeier, G.: Editor: K.U. Kainer (Hrsg.), Magnesium, Wiley-VCH, Weinheim (2009) 1357.
[4] Grundmeier, G.; Janke, S.; Ozcan, O. and Birkenheuer, S.: “Surface and thin film engineering of zinc alloy coated steel sheets” Galvatech 2011, 21-24 June 2011
[5] Wapner, K.; Stratmann, M. and Grundmeier, G.: International Journal of Adhesion & Adhesives 28 (2007) 59.
[6] Giza, M.; Thissen, P. and Grundmeier, G.: Langmuir 24 (2008) 8688.
[7] Perdew, J.P.; Burke K. and Ernzerhof, M.: Phys. Rev. Lett., 77 (1996) 3865.
[8] JSoler, J.M.; Artacho, E.; Gale, J.D.; García, A.; Junquera, J.; Ordejón P. and Sánchez-Portal, D.: Phys.: Condens. Matter, 14 (2002) 2745.
[9] Scandolo, S.; Giannozzi, P.; Cavazzoni, C.; de Gironcoli, S., Pasquarello, A. and Baroni, S.: Kristallogr., 220 (2005) 574.
[10] Troullier, N. and Martins, J.L.: Phys. Rev. B, 43 (1991) 1993.
[11] Louie, S.G., Froyen, S. and Cohen, M.L.: Phys. Rev. B, 26 (1982) 1738.
[12] Kleinman, L. and Bylander, D.M.: Phys. Rev. Lett., 48 (1982) 1425.
[13] Coquet, R.; Hutchings, G.J.; Taylor, S.H., Willock, D.J. and Mater, J.: Chem., 16 (2006) 1978.
[14] Scamehorn, C.A.; Harrison, N.M. and McCarthy, M.I.: J. Chem. Phys. 101 (1994) 1547.
[15] Langel, W.: Surf. Sci. 496 (2002) 141.
[16] Lazarov, V.K.; Cai, Z.; Yoshida, K., Zhang, K. H. L.; Weinert, M.; Ziemer, K.S. and Hasnip, P.: PRL 107 (2011) 056101

6.10 Investigating the thermal and enzymatic taxifolin-alphitonin rearrangement

Project coordinator	Professor Dr. Michael Gütschow, Pharm. Inst., University of Bonn
Project members	Prof. Dr. Paul W. Elsinghorst, Pharm. Inst., University of Bonn

General Problem Description

The project addresses issues of tautomerism and isomerism and their contribution to small molecule-protein interactions. While tautomerism reflects the possibility of one molecule to exist in more than one constitution with respect to where hydrogen atoms are present within their structure, isomerism gives rise to more than one spatial structure of a molecule that may or may not be interconvertible at ambient temperature. The interaction of small molecules, *e.g.,* endogenous substances, pharmaceuticals, or toxins, with their target proteins, *e.g.,* enzymes or receptors, is – from a chemical point of view – basically the result of an energetically favorable complex of the two. Their interaction is driven by a complex interplay related to hydrophobic, electrostatic, or dipole-dipole attractions and hydrogen bonds.

During the last two decades, medicinal chemistry has developed sophisticated *in silico* techniques to explore small molecule-protein interactions from a theoretical point of view. What is nowadays referred to as molecular docking or molecular dynamics plays a key role in modern drug development. The available software tools basically take the three-dimensional structure of a small molecule and a protein and search for a favorable complex by twisting/bending their structures according to chemical and/or physical laws. The results' accuracy largely depends on the quality of the input structures.

With respect to tautomerism, some of the available software tools can generate possible tautomers of one structure but usually do not consider their energetic likeliness. Thus, comparing theoretically possible tautomers with regard to their intrinsic energy states and interconvertability at room temperature will probably rule out some of them. Isomerism can also be temperature-dependent. Bonds within a molecular structure may cause rotational barriers, *i.e.,* the parts of both ends of a bond may not rotate freely around their connecting bond because of spatial clashes between them. Rotational barriers may be overcome by twisting/bending the molecular structure at a certain energy cost. If the room temperature cannot afford this energy cost, the resulting two structures are referred to as atropisomers and need to be considered separately.

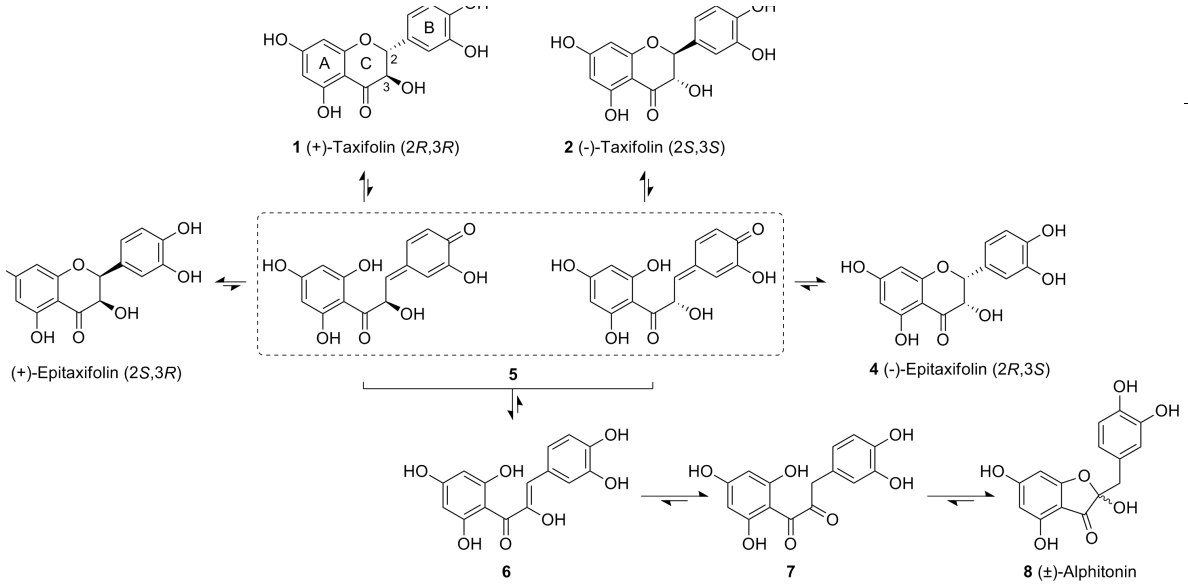
Quantum chemical software packages include suitable techniques to assess the energetic state associated with possible tautomers/isomers of small molecules. Their high demand for computational resources limits the use on standalone PCs and makes them predestined for HPC systems. Two well established software packages (GAUSSIAN03, ORCA) were used in this project in combination with other *in silico* methods and experimental data.

Problem details and work done

A. (see also ref. 1)

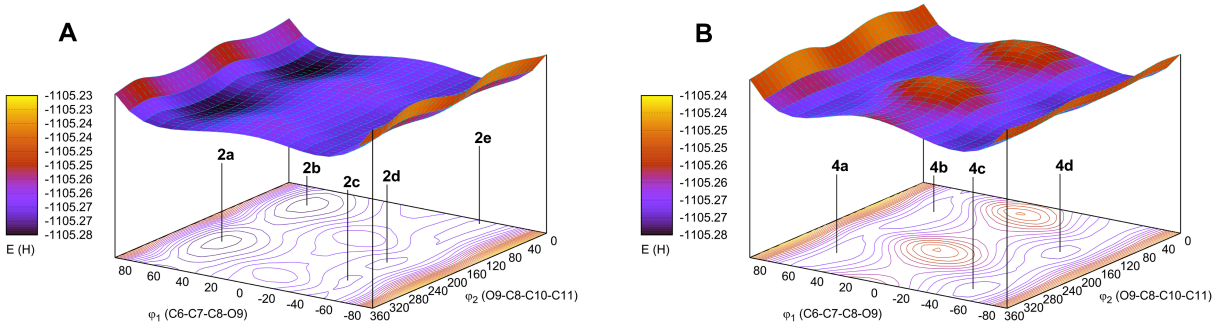
Both, taxifolin and alphitonin, appear as native constituents of plants and were previously described as quercetin metabolites of intestinal bacteria. Taxifolins refer to the *trans*-configured diastereomers of dihydroquercetin, whereas the *cis*-configured ones are called epitaxifolins (Figure 1). Ring-opening and successive ring-closure through two intermediate quinone methides **5** connects all diastereomers **1**-**4**. All dihydroquercetin diastereomers may further tautomerize through intermediates **6**, **7** followed by a ring contraction to produce alphitonin (**8**), a benzofuranone. This reaction can either be achieved at elevated temperature or through the action of a partially purified enzyme preparation from *Eubacterium ramulus*.

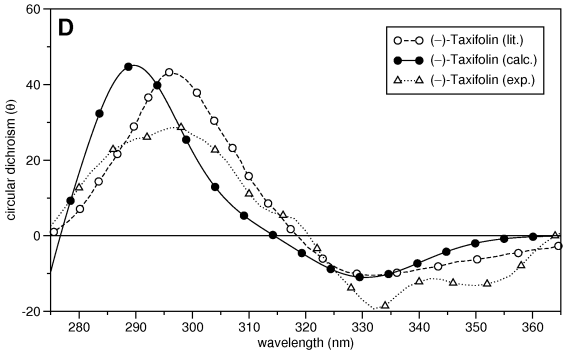
This project investigates the thermal and enzymatic conversion from (±)-taxifolin (**1**, **2**) to alphitonin (**8**). Chromatographic separation of the four dihydroquercetin diastereomers **1**-**4** in combination with circular dichroism spectroscopy permitted the elucidation of the kinetics of this rearrangement and a characterization of the different reaction pathways involved.

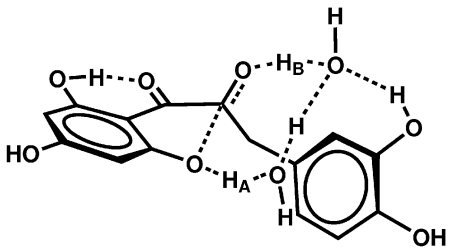


ɟure 1: Rearrangement of dihydroquercetins (1-4) into alphitonin (8) via the quinone methides 5, the ɛalcone 6, and the diketone 7.

To elucidate the conversion of taxifolin into alphitonin on a molecular basis, we established a chiral HPLC system capable of separating all taxifolin diastereomers. Theoretically predicted CD spectra were acquired to assign each structure to the CD recording from the HPLC. Therefore, minimum energy conformers were obtained from PES scans simulating a full rotation of the B–C ring connecting bond (Figure 2, top). Corresponding CD spectra were weighted by Boltzmann statistics and compared to literature assignments (Figure 2, bottom). In addition, a transition state **9** involving two water molecules for proton relay was identified, which is responsible for on-column racemization of alphitonin, which in turn does not allow enantiospecific separation of the two putative alphitonin isomers.







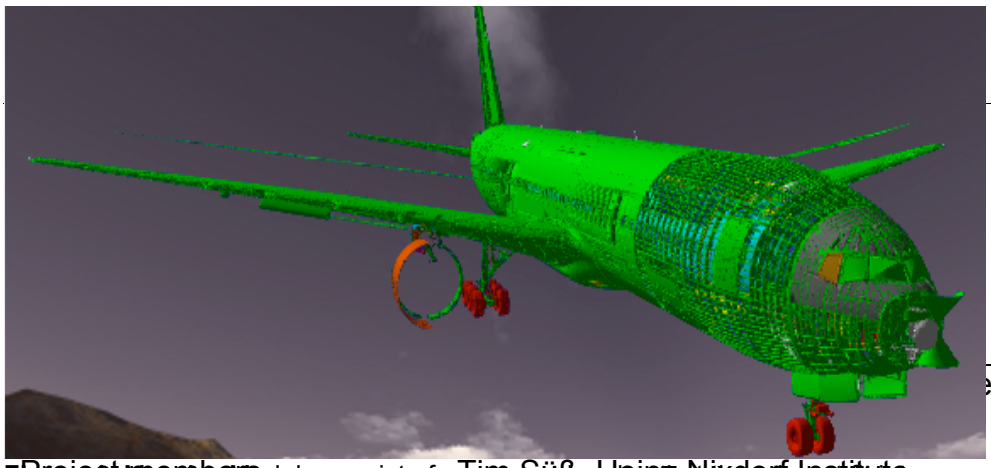
9

Resource Usage

cluster: Arminius, software: Gaussian, Orca

References

[1] Elsinghorst, P.W.; Cavlar, T.; Müller, A.; Blaut, M.; Braune, A. and Gütschow, M.: The thermal and enzymatic taxifolin–alphitonin rearrangement. *J. Nat. Prod.* **2011**, *74*, 2243-2249.



er Heide,

Project members: Tim Süß, Heinz Nixdorf Institute

Figure 1: Massive models consist of so many triangles that they do not fit into a node's primary memory. This model of a Boeing 777 consists of approximately 350,000,000 triangles and requires more than 8\,GiB memory.

General Problem Description

Complex polygonal 3D models may consist of hundreds of millions of triangles and require multiple gigabytes of memory (see Figure 1.1). Rendering such Massive Models in real-time is one of the most challenging problems in modern computer graphics [KBF05]. A user should be able to navigate through a scene or through models interactively while at least six to ten frames per second are computed. The parallelization of the rendering process is a common approach to this problem [KDG+08]. Many real-time rendering algorithms can improve performance by distributing the load among multiple computers. For each frame that is displayed, an image for the current camera position must be rendered. To produce images of polygonal 3D models, their geometric primitives are usually sent through a rendering pipeline, where they are transformed into pixels [AMHH08]. The parallelization of such real-time, pipeline rendering algorithms is rarely done because PC clusters completely equipped with modern graphic adapters are still rare. Usually, PC clusters are intended to be used for other applications, such as scientific computations.

The Paderborn Center of Parallel Computing's (PC²) Arminius PC cluster is equipped with weak graphics adapters and a fast network via Infiniband. Additionally, PC² offers a few nodes equipped with powerful, high end graphics cards. However, this kind of cluster can hardly be used for standard parallel pipeline rendering techniques as proposed by Molnar et al. [MCEF94, MCEF08]. Typically, the performance of these methods depends on the slowest node. Due to these reasons, we put the focus of our research on the development of new parallel pipeline rendering algorithms for such heterogeneous PC clusters. We require that these heterogeneous systems include a small group of powerful visualization nodes and a large group of weaker back-end nodes. While the visualization nodes should be equipped with high end graphics adapters, the back-end nodes only require a weak graphics performance. The back-end nodes should be equipped with common hardware, and a network must connect the different nodes. Our objective is to render complex 3D scenes in real-time, using such heterogeneous environments.

On PC²'s Arminius cluster, we developed two different parallel out-of-core rendering systems [SWF10a, SWF10b, SKJ+11]. The rendering method of these rendering systems is approximate, which means that the final generated images contain (few) pixel errors. In these parallel out-of-core systems, the weak back-end nodes serve as secondary memory to the visualization nodes. The complete scene is distributed among these weak nodes and stored in their primary memory to allow a fast data access. When scene objects are requested, the back-end nodes test the visibility of these objects instead of sending them blindly.

Our first parallel out-of-core rendering system uses a version of the *c*-load-collision protocol to balance the rendering load and the nodes' contention. The back-end nodes perform visibility tests while they only have access to a subset of all objects and to the global, but aged, distance information of the other objects. Due to the aged distance information, the back-end nodes cannot guarantee a determination of all visible objects until the information are updated. The positively tested objects are sent to the visualization node, where they are rendered and displayed.

The second parallel out-of-core rendering system combines a self-developed spatial, hierarchical data structure, the so-called hull tree [SKJF11], with another data structure, the so-called randomized sample tree. The hull tree covers a scene's objects more tightly than other commonly used data structures. Additionally, we store an approximation of each object to improve and accelerate the visibility test. Our associated approximate rendering algorithm exploits this structure. Each back-end node stores a small subset of the original objects and approximations for the other objects. The back-end nodes perform visibility tests with this mix of originals and approximations, which are organized in a hull tree.

In these parallel rendering systems, we were confronted with the problem of data distribution, the problem of evenly distributing the computational load, the problem of providing information to perform suitable visibility tests, and the need of reducing the number of objects that are sent across the network.

We solved these problems as followed:
Due to a randomized distribution, we achieve a good load balancing in both parallel out-of-core rendering systems. In our first system, we achieved suitable visibility tests if we used global, but aged, distance information of a scene's objects. The data management protocol of this system achieves a good balancing of the load.
In our second system, the hull tree reduces the complexity of the visibility tests. Due to the hull tree's combination with a randomized sample tree, sending the visible objects across the network is distributed among multiple frames.
In both systems, we tested objects' visibility a priori on the back-end nodes instead of sending them blindly. These tests reduced the network load. Thus, we could reduce the network requirements for the second parallel out-of-core rendering system.

References

[1] Akenine-Möller, T.; Haines, E. and Homan, N.: *Real-Time Rendering 3rd Edition.* A. K. Peters, Ltd., Natick, MA, USA, 2008.

[2] Kasik, D.J.; Buxton, W. and Ferguson, D.R.: *Ten CAD challenges.* IEEE Computer Graphics and Applications, 25:81-92, March 2005.

[3] Kasik, D.; Dietrich, A., Gobbetti, E.; Marton, F.; Manocha, D.; Slusallek, P.; Stephens, A. and Yoon, S.: *Massive model visualization techniques: course notes.* In ACM SIGGRAPH 2008 classes, SIGGRAPH '08, pages 40:1-40:188, New York, NY, USA, 2008. ACM.

[4] Molnar, S.; Cox, M.; Ellsworth, D. and Fuchs, H.: *A sorting classification of parallel rendering.* IEEE Computer Graphics and Applications, 14:23-32, July 1994.

[5] Molnar, S.; Cox, M.; Ellsworth, D. and Fuchs, H.: *A sorting classification of parallel rendering.* In ACM SIGGRAPH ASIA 2008 courses, SIGGRAPH Asia '08, pages 35:1-35:11, New York, NY, USA, 2008. ACM.

[6] Süß, T.; Koch, C.; Jähn, C.; Fischer, M. and Meyer auf der Heide, F.: *Ein paralleles Out-of-Core Renderingsystem für Standard-Rechnernetze.* In Jürgen Gausemeier, Michael Grafe, and Friedhelm Meyer auf der Heide, editors, Augmented & Virtual Reality in der Produktentstehung, volume 295 of HNI-Verlagsschriftenreihe, Paderborn, pages 185-197. Heinz Nixdorf Institut, Universität Paderborn, May 2011.

[7] Süß, T.; Koch, C.; Jähn, C. and Fischer, M.: *Approximative occlusion culling using the hull tree.* In Proceedings of the Graphics Interface 2011, pages 79-86. Canadian Human-Computer Communications Society, May 2011.

[8] Süß, T.; Wiesemann, T. and Fischer, M.: *Evaluation of a c-load-collision-protocol for load-balancing in interactive environments.* In Proceedings of the 5th IEEE International Conference on Networking, Architecture, and Storage, pages 448-456. IEEE Computer Society, IEEE Press, July 2010.

[9] Süß, T.; Wiesemann, T. and Fischer, M.: *Gewichtetes c-Collision-Protokoll zur Balancierung eines parallelen Out-of-Core-Renderingsystems.* In Jürgen Gausemeier and Michael Grafe, editors, Augmented & Virtual Reality in der Produktentstehung, HNI Verlagsschriftenreihe, Paderborn, pages 39-52. Universität Paderborn, HNI Verlagsschriftenreihe, Paderborn, 2010.

6.12 Shape Optimizing Load Balancing for Parallel Adaptive Numerical Simulations

Project coordinator	Prof. Dr. Burkhard Monien, University of Paderborn
Project members	Jun.-Prof. Dr. Henning Meyerhenke, Karlsruhe Institute of Technology
Supported by	DFG (SPP 1307 Algorithm Engineering)

General Problem Description

Numerical simulations are very important tools in science and engineering for the analysis of physical processes. Application areas include fluid dynamics, structural mechanics, nuclear physics, and many others [1]. Usually the simulation domain is discretized into a mesh, which can be regarded as a graph with geometric (and possibly other) information. The discretization transforms the equations governing the simulated process into large linear systems. When these systems are solved in parallel by iterative numerical solvers, the mesh elements must be distributed evenly onto the processors of the parallel system. This is due to the fact that the elements represent the computational load, which should be balanced for efficiency. Moreover throughout the solution process, neighboring elements of the mesh need to exchange values during each iteration to update their own value. Due to the high costs of inter-processor communication, neighboring mesh elements should reside on the same processor.

A good initial assignment of subdomains to processors can be found by solving the graph partitioning problem (GPP) [11]. The most common GPP formulation for an undirected graph G = (V, E) asks for a division of V into k pairwise disjoint subsets (*parts*) such that all parts are no larger than $(1+\varepsilon)\cdot\lceil |V|/k \rceil$ (for small $\varepsilon \geq 0$) and the *edge-cut*, i. e., the total number of edges having their incident nodes in different subdomains, is minimized.

In many numerical simulations some areas of the mesh are of higher interest than others. For instance, during the simulation of the interaction of a gas bubble with a surrounding liquid, one is interested in the conditions close to the boundary of the fluids. Another application, among many others, is the simulation of the dynamic behavior of biomolecular systems [1]. To obtain an accurate solution, a high resolution of the mesh is required in the areas of interest. To use the available memory efficiently, one has to work with different resolutions in different areas. Moreover, the areas of interest may change during the simulation, which requires adaptations in the mesh and may result in undesirable load imbalances. Hence after the mesh has been adapted, its elements need to be redistributed such that every processor has a similar computational effort again. While the balance

objective can be met by solving the GPP again, the repartitioning process needs to find new partitions of high quality. Additionally, as few nodes as possible should be moved to other processors since this *migration* causes high communication costs and changes in the local mesh data structure.

The most popular graph partitioning and repartitioning libraries use local node-exchanging heuristics, like Kernighan-Lin (KL), within a multilevel improvement process to quickly compute solutions with low edge cuts [11]. Yet, their deployment can have certain drawbacks. First of all, minimizing the edge-cut with these tools does not necessarily mean minimizing the total runtime of parallel numerical simulations [4]. The total communication volume can be minimized by hypergraph partitioning [2]. However, synchronous parallel applications need to wait for the processor with the longest computing time. Hence, the *maximum norm* (i. e., the worst part in a partition) of the simulation's communication costs is of higher importance. Moreover for some applications, the shape of the subdomains plays a significant role. Optimizing partition shapes, however, requires additional techniques (see [9] and the references therein), which are far from being mature. Finally due to their sequential nature, the most popular repartitioning heuristics are not easy to parallelize—although significant progress has been made [5].

Problem details and work done

Our partitioning algorithm DibaP aims at computing well-shaped partitions and uses disturbed diffusive schemes to decide not only how many nodes move to other parts but also *which* ones. It is inherently parallel and overcomes many of the above-mentioned difficulties, as could be shown experimentally for static graph partitioning [9]. While it is much slower than state-of-the-art partitioners, it often obtains better results.

With the work performed in the reporting period, we further explore the disturbed diffusive approach with the focus on repartitioning for load balancing. First we have extended the implementation of DibaP for MPI parallel repartitioning, yielding PDibaP. With this implementation we have performed various repartitioning experiments with dynamic sequences of benchmark graphs, whose sizes range between 1M and 16M vertices. These experiments are the first using PDibaP for repartitioning and show the suitability of the disturbed diffusive approach. In the following we summarize our findings. More details can be found in [7].

We have conducted our experiments on the Arminius cluster of the Paderborn Center for Parallel Computing. The values measuring the communication volume of an underlying linear solver show that PDibaP consistently computes the best partitions. With about 12–19% improvements on parallel Jostle and about 34–53% on ParMETIS, the advancement is clearly higher than the approximately 7% obtained for static partitioning [9]. This higher

improvement is due to the fact that parallel KL (re)partitioners often compute worse solutions than their serial counterparts for static partitioning. The results for the migration volume are not consistent. All tools have a similar amount of best values. While PDibaP has a more constant migration volume over time within the same dynamic graph sequence, the values for parallel Jostle and ParMETIS show a higher amplitude. It depends on the instance which strategy pays off. These results lead to the conclusion that PDibaP's implicit optimization focuses more on good partitions than on small migration costs. In some cases the latter objective should receive more attention. As currently no explicit mechanisms for migration optimization are integrated, such mechanisms could be implemented if one finds in other experiments that the migration costs become too high with PDibaP.

The runtime of the tools for the dynamic graph instances used in this study can be characterized as follows. ParMETIS is the fastest, taking from a fraction of a second up to a few seconds for each repartitioning step. Parallel Jostle is by approximately a factor of 2-3 slower than ParMETIS. PDibaP is significantly slower than both tools, with an average slowdown factor of about 25-50 compared to ParMETIS. It requires from a few seconds up to a few minutes for each repartitioning step.

It needs to be stressed that a high repartitioning quality is often very important. Usually, the most time consuming parts of numerical simulations are the numerical solvers. Hence, a reduced communication volume provided by an excellent partitioning can pay off unless the repartitioning time is extremely high. Nevertheless, a further acceleration of shape-optimizing load balancing is of utmost importance and an aspect of further investigation.

We would like to thank the Paderborn Center for Parallel Computing for providing the hardware necessary to perform the experiments in this study.

References

[1] Baker, N.A.; Sept, D.; Holst, M.J. and McCammon, J.A.: The adaptive multilevel finite element solution of the Poisson-Boltzmann equation on massively parallel computers. IBM J. of Research and Development, 45(3.4):427 –438, May 2001.

[2] Catalyurek, U. and Aykanat, C.: Hypergraph-partitioning-based decomposition for parallel sparse-matrix vector multiplication. IEEE Transactions on Parallel and Distributed System, 10(7):673–693, 1999.

[3] Fox, G.; Williams, R. and Messina, P.: Parallel Computing Works! Morgan Kaufmann, 1994.

[4] Hendrickson, B. and Kolda, T.G.: Graph partitioning models for parallel computing. Parallel Comput., 26(12):1519–1534, 2000.

[5] Holtgrewe, M.; Sanders, P. and Schulz, C.: Engineering a scalable high quality graph partitioner. In 22nd International Parallel and Distributed Computing Symposium (IPDPS 2010), pages 1–12. IEEE, 2010.

[6] Meyerhenke, H.: Dynamic load balancing for parallel numerical simulations based on repartitioning with disturbed diffusion. In Proc. Internatl. Conference on Parallel and Distributed Systems (ICPADS'09), pages 150–157. IEEE Computer Society, 2009.

[7] Meyerhenke, H.: Shape Optimizing Load Balancing for Parallel Adaptive Numerical Simulations Using MPI. Accepted for presentation at 10th DIMACS Implementation Challenge Workshop.

[8] Meyerhenke, H. and Monien, B.: On Multilevel Diffusion-based Load Balancing for Parallel Adaptive Numerical Simulations. Presented at SIAM Conference on Computational Science and Engineering (CSE'11), Reno, (Nevada, USA), February/March 2011.

[9] Meyerhenke, H.; Monien, B. and Sauerwald, T.: A new diffusion-based multilevel algorithm for computing graph partitions. Journal of Parallel and Distributed Computing, 69(9): 750–761, 2009. Best Paper Awards and Panel Summary: IPDPS 2008.

[10] Meyerhenke, H.; Monien, B. and Schamberger, S.: Graph partitioning and disturbed diffusion. Parallel Computing, 35(10–11):544–569, 2009.

[11] Schloegel, K.; Karypis, G. and Kumar, V.: Graph partitioning for high performance scientific simulations. In The Sourcebook of Parallel Computing, pages 491–541. Morgan Kaufmann, 2003.

6.13 Solution of a large scale inverse electromagnetic scattering problem

Project coordinator	Prof. Dr. Andrea Walther, Institute of Mathematics, University of Paderborn
Project members	Maria Brune, Institute of Mathematics, University of Paderborn
Supported by	BMBF

General Problem Description

This project focuses on the solution of a large scale inverse electromagnetic scattering problem where a 3D reconstruction of electromagnetic properties in a large cubic area should be computed [1]. Since there is only one source available, the corresponding forward problem comprises an instationary 3D simulation. The propagation of the electromagnetic field is governed by the time dependent Maxwell's equations. The simulated data is discretized by a FDTD (Finite Differences in Time Domain) method. Our aim is to reconstruct a domain, which is divided into up to 1000^3 grid cells. Due to the huge amount of data, high performance methods are indispensable. Therefore, several parallelization techniques are incorporated.

Currently, we want to reconstruct the permittivity. However, it is also possible to reconstruct other electromagnetic properties. For this purpose, we minimize the discrepancy between the simulated and the measured data. The arising optimization problem is solved through the limited-memory quasi-Newton algorithm l-BFGS in its bounded version. The derivatives are provided by the algorithmic differentiation tool ADOL-C, which is developed and maintained in our working group.

As these optimization problems are usually ill-posed, one has to apply suitable regularization approaches. Another important issue is the detection of a global minimum since we have a multi-modal domain, and the correct parameter distribution only corresponds to the global minimum.

Problem details and work done

One important part of the project is the discretization of the time dependent Maxwell's equations that are described as follows

$$\frac{-\partial B}{\partial t} = rot\, E$$

$$\frac{\partial D}{\partial t} = rot\, H - J$$

The material equations represent the relation between the flux densities and the field strength and can be formulated for the electrical case as

$$D = \epsilon E \quad \epsilon = \epsilon_0 \epsilon_r$$

where $\in$ denotes the permittivity, $\in_0$ denotes the permittivity of vacuum, and $\in_r$ the relative permittivity. Hence, we want to reconstruct the parameter $\in_r$. For the discretization of the Maxwell's equations, the FDTD-method is applied. This approach is based on the discretization of the curl equations. The components of the electrical field E and the magnetic field H in particular are staggered in time and space, i.e., the leapfrog scheme is used.

The objective function can be formulated as follows

$$j(\epsilon) = \sum_{(i,j,k)\in M} \sum_{n=0}^{N} \left(\frac{1}{2}\|u(\epsilon) - u^{obs}\|^2\right) + \frac{\beta}{2} R(\epsilon)$$

where $u(\in)$ denotes the simulated state, u^{obs} denotes the observed state, $R(\in)$ the penalty term, and β the regularization parameter. For the regularization, we applied different well-established approaches: Tikhonov-Regularization, Total-Variation-Regularization, and an approach that uses the squared gradient norm as a penalty term. This optimization problem is solved by the l-BFGS-b algorithm that also requires first order derivatives to approximate second order derivatives. The update of the inverse Hessian is performed by a limited variation of the well-known BFGS-Update. For the calculation of the derivatives, we employ ADOL-C. In our case, we use the reverse mode because then the complexity of the gradient computation does not depend on the number of independent variables. Here, we also incorporated checkpointing strategies to cope with the high memory requirement postulated by the reverse mode.

For the numerical tests, we discretize the domain into 100^3 grid cells. The target domain that we want to reconstruct is shown in Illustration 1. We primarily want to reconstruct

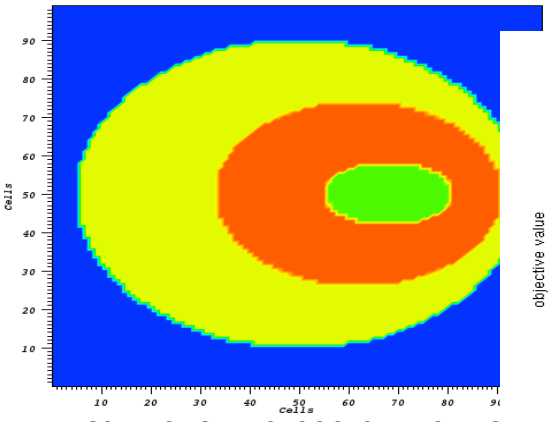


Illustration 1: Target domain

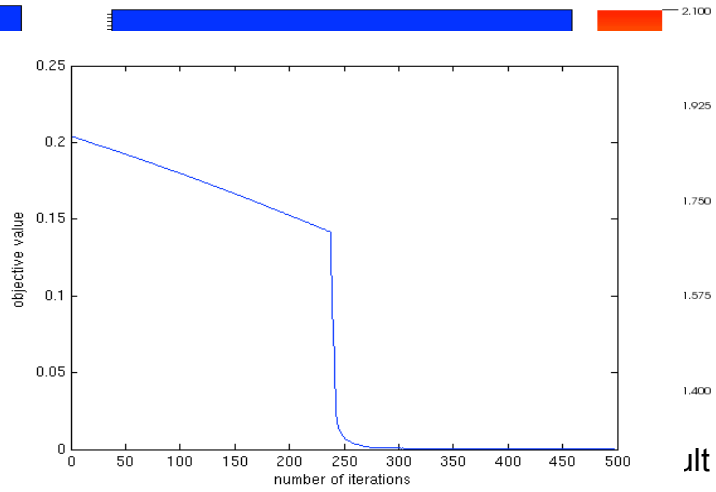


Illustration 3: Reduction of objective value

01

do not
osen as
Matched
ver, we
ne huge
ult after 490

amount of data, we employe

iterations. One can clearly recognize the shape of the inner ellipses.

Illustration 3 exemplifies a typical behavior of the applied optimization algorithm. During the first 250 iterations, no sufficient decrease can be achieved. However from this point on, the optimizer finds a good descent direction, and it converges rapidly.

Part of our future work is to incorporate a suitable globalization strategy to make sure that we obtain the global minimum.

Resource Usage

The optimization runs reported above were performed on the HPC System Arminius. It is necessary to compute on a HPC System so that we can use the compute power to handle the huge amount of data arising from the application. We usually start optimization runs two or three times a week.

References

[1] Landmann, D.; Plettemeier, D.; Statz, C.; Hoffeins, F.; Markwardt, U.; Nagel, W.; Walther, A.; Herique, A. and Kofman, W.: Three-dimensional reconstruction of comet nucleus by optimal control of Maxwell's equations: A contribution to the experiment CONSERT onboard space mission ROSETTA. Proceedings IEEE International Radar Conference 2010, pp.\,1392-1396 (2010)

6.14 Optimization of optimal power flow problems in alternating current networks

Project coordinator	Prof. Dr. Andrea Walther, Institute of Mathematics, University of Paderborn
Project members	Maria Brune, Institute of Mathematics, University of Paderborn
Supported by	Réseau de Transport d`Electricité

General Problem Description

This project is a joint work with RTE France, one of the most important transmission system operators in Europe. They are responsible for the operation, maintenance, and development of the electricity network in France. In this project we are interested in the optimization of power flow problems. These problems typically arise in the context of optimization and secure exploitation of electrical power in alternating current (AC) networks [1]. Since there are several contingencies within the network, a very important issue is the improvement of the reliability of power supply. Therefore, an electrically secured steady state, while considering hypothetic electrical failures, has to be determined.

The optimization algorithm estimates the state variables when certain contingencies are given. In general we have to solve a nonlinear optimization problem with at least two constraints at each node of the network. Usually, optimal power flow problems are solved through interior-point methods. In this project we applied the well-established interior point optimizer IPOPT coupled with the algorithmic differentiation tool ADOL-C, which is developed and maintained in our working group.

Due to the large scale of the problem even at the base case, the optimization time increases rapidly with the number of contingencies. Therefore, suitable parallelization strategies are required. A practical approach is to run recurring calculations within the optimization routine in parallel as far as possible. Parallelized computations of first and second order derivatives are also desirable. Therefore, we coupled a new version of ADOL-C with our simulation code. This new version of ADOL-C provides parallelized functions for the computation of first and second order derivatives.

Problem details and work done

In this project we particularly focus on the optimization and efficient computation of recurring evaluations of the objective function and the derivatives.

The optimization aims at the determination of an optimally secured steady state when several contingencies are known. The op as a least square problem and looks as follows

$$h(x) = 0$$

where x is a bound vector that consists o e variables that are not directly controllable, and the other part contains the control variables that are directly controllable. The control variables are, e.g., fictive active and reactive injections at each node. The state variables are voltage magnitude and angle.

Due to Kirchhoff's law, the numerous equalities represent the active and reactive balance of each bus in the network. The inequalities assure realistic estimations of new state variables according to the power limitations of the production units.

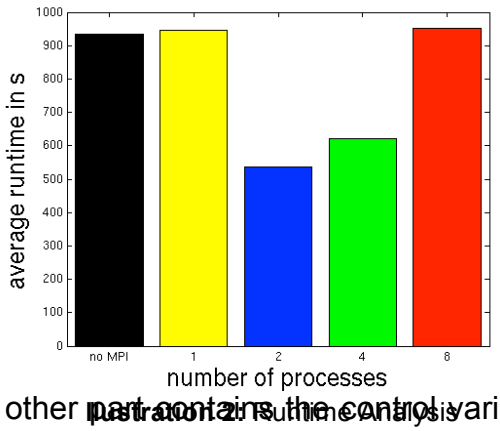


Illustration 2: Runtime Analysis

A challenging task is to solve the large scale problem efficiently because even the network with few contingencies is already large-scaled, and consequently the problem size increases linearly with the number of contingencies. To cope with this problem, high performance strategies are required. Therefore, we use the interface MPI (Message Passing Interface) and parallelize recurring computations, like the evaluation of the objective function, as well as equalities because they have to be computed at least once in one iteration. The simulation code, which was provided by RTE, was written in C++ and existed only in a sequential version. For the parallelization the whole structure of the program had to be modified. As the optimization problem is solved by a sequential optimizer, only one process, the root process, accesses the optimization routine. It is responsible for the distribution of the current state vector and the collection of the computed results by the other processes.

For the runtime analysis, we perform several tests on the HPC-System Arminius. For one test case we take a network with 8937 transmission nodes, 17874 constraints, and 15127 electric lines. There are 35748 independent and 17874 dependent variables. In Illustration 1 the average runtime for the optimization of a varying number of processes is shown. In this

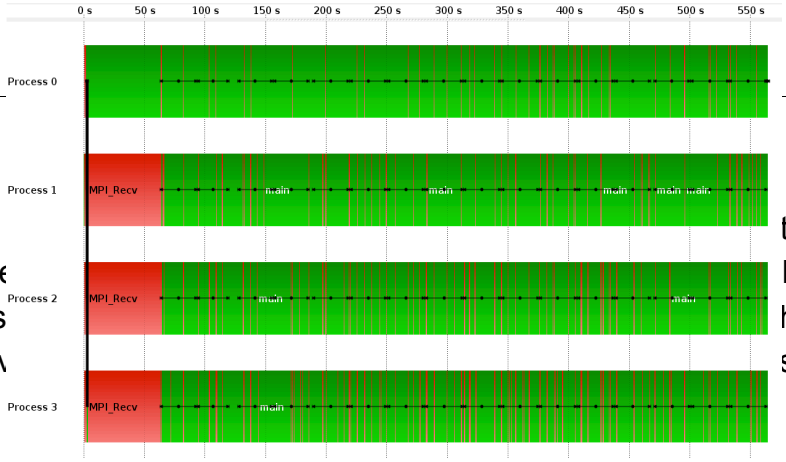


Illustration 3: Processchart (by Vampir)

figure one can see that test runtime. Compared to the seque he usage of four or more processes his might be the relatively short ov s shown in Illustration 2.

Especially the beginning of the optimization takes a lot of time because only the root process enters the optimization routine and needs to finish the initializations first, so the other processes have to wait (see Illustration 2). This problem can be solved by applying a parallel optimization routine. To couple this program with a parallel optimizer is part of our future work. Another important issue is the improvement of the communication between the processes as well as the improvement of the load balancing.

Resource Usage

The optimization runs reported above were performed on the HPC System Arminius. The purpose of computing on the HPC-System was to test MPI-parallelized software. During the project we started weekly computations on that system.

References

[1] Brune, M.; Castaing, L. and Walther, A.: Optimization of Optimal Power Flow Problems. Proceedings in Applied Mathematics and Mechanics 2011, will be published at the end of 2011

6.15 Computational studies on lactide polymerization with zinc guanidine complexes (Case c – hardware users)

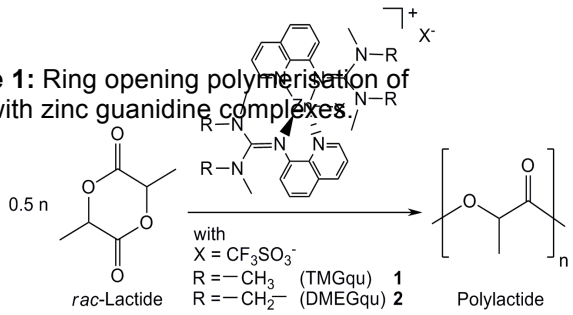
Project coordinator	Prof. Dr. Sonja Herres-Pawlis, Technische Universität Dortmund / LMU München
Project members	Anton Jesser, Technische Universität Dortmund
Supported by	BMBF – MoSGrid and DFG

General Problem Description

Polylactide (PLA) is an aliphatic polyester which can be produced by ring-opening polymerisation (ROP) of lactide (LA). PLAs have proven to be the most attractive and useful class of biodegradable polyesters starting to conquer a billion-dollar-market in the substitution of petrochemical plastics.[1] Hence, the development of new single-site metal catalysts for the ROP of lactide has seen tremendous growth over the past decade.[2] A vast multitude of well-defined Lewis acid catalysts following a coordination-insertion mechanism has been developed for this reaction mainly based on tin, zinc, aluminium and rare earth metals.[1,2]

However, the high polymerisation activity of all these systems is often combined with high sensitivity towards air and moisture. For industrial purposes and especially the breakthrough of PLA in the competition with petrochemical-based plastics, there is an exigent need for active initiators that tolerate air, moisture and small impurities in the monomer.[1] The disadvantageous sensitivity can be ascribed to the anionic nature of the ligand systems stabilising almost all of these complexes. Up to now, only few ROP active systems using neutral ligands in single-site metal catalysts have been described. They make use of strong donors such as guanidines[3] and phosphinimines.[4] Guanidine systems gain their unique donor properties from the ability to effectively delocalise a positive charge over the CN_3 moiety.[3b] The guanidine ligands represent strong donors comparable to b-ketiminates but the resulting zinc complexes possess a considerably higher stability towards moisture and lactide impurities. Previous studies have demonstrated that the bis(chelate) systems $[Zn(TMGqu)_2OTf]OTf$ (1) and $[Zn(DMEGqu)_2OTf]OTf$ (2) exhibit great robustness and high catalytic activity at the same time.[3b] However, the question for the proceeding mechanism for this special catalyst class which is active without the presence of alkoxides or alcohols remained open. Herein we report on an integrated study on the mechanism for bis(chelate) guanidine complexes combining extensive kinetic analyses, spectroscopic studies and DFT calculations.

Problem details and work done



Scheme 1: Ring opening polymerisation of lactide with zinc guanidine complexes.

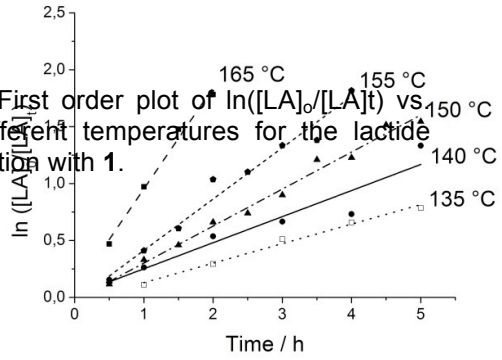


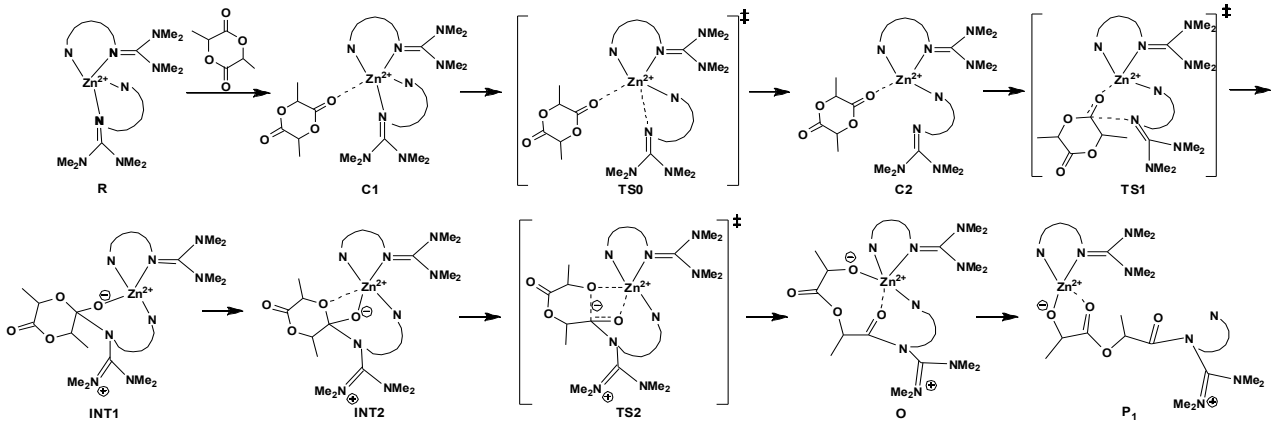
Figure 1: First order plot of ln([LA]$_o$/[LA]t) vs. time at different temperatures for the lactide polymerisation with **1**.

The polymerisation of rac-LA with the zinc complexes **1** and **2** was performed under bulk conditions without the presence of additional co-initiators at elevated temperature for reasons of industrial applicability (Scheme 1).[3] The inspection of the molecular weights M_n reveals a linear relationship between M_n determined by GPC and conversion monitored by NMR spectroscopy. The comparison between the experimental M_n values and the calculated ones ($M_{n,theo}$) clearly shows that the former are significantly higher. Kinetic measurements for polymerisations with **1** were carried out between 135 and 165°C (Figure 1). The semilogarithmic relationship between the decrease in LA concentration and reaction time shows the living character of the polymerisation as the number of growing chains remains constant (no irreversible chain transfer and/or termination reaction).[1] From an Eyring plot, the activation parameters could be derived (DH‡ = 79(4) kJ mol^{-1} and DS‡ = -33(4) J K^{-1} mol^{-1}) which are in good accordance with values reported by Okuda.[5] They indicate an ordered transition state typical for a coordination-insertion mechanism.

With constant rac-LA concentration [LA] and variable concentration of **1**, the polymerisation is first order in [**1**], resulting in an overall second-order rate law for LA consumption d[LA]/dt = k_p[1][LA] with a value for the polymerisation rate k_p of 2.6•10^{-3} s^{-1} M^{-1}. Inspired by these polymerisation characteristics, we propose, analogously to classical metal alkoxide single-site initiators, that in the zinc guanidine initiator systems **1** and **2** the basic guanidine function acts as nucleophilic ring-opening agent. To support this hypothesis we tried to identify the end group by spectroscopic techniques. Energy dispersive X-ray analysis (EDX) gave within the limit of detection no zinc signal in the worked-up polymer. Thus, the zinc complex does not remain as chain end. Due to the intense yellow colour of the ligands, their complexes and the obtained polymers, we performed UV/Vis and fluorescence measurements. The observed absorption features in the UV range are not present in samples produced using tin octanoate. The guanidine ligands themselves have intense absorptions at 347 nm (8800 L mol^{-1} cm^{-1}) and 371 nm (1320 L mol^{-1} cm^{-1}) for DMEGqu and TMGqu, respectively.[6] This π-π* transition of the aromatic quinoline system can be traced in the polymer. Remarkably, the distinct intensity

difference of the two ligands shows up in the UV absorption spectra of the corresponding polymer samples as well. In the fluorescence spectra of ligands and polymers, the emission of the ligands appears at 500 (DMEGqu) and 510 nm (TMGqu). Notably, the DMEGqu ligand shows the double fluorescence intensity compared to TMGqu. This intensity difference can be retrieved in the fluorescence of the solid-state emission spectra of the PLA samples as well: The samples produced by using DMEGqu complex **2** emit with double intensity compared to those obtained by the TMGqu complex **1**.

In order to exclude the possibility that ligand is merely co-precipitated with the polymer, PLA samples were worked up in several cycles of dissolution in CH_2Cl_2, precipitation in ethanol and drying. After each step samples for the fluorescence studies were taken. Almost no loss in fluorescence intensity was detected indicating that the ligand is retained in the polymer. In an additional experiment the fluorescence intensity of samples with different molecular weight was investigated. Samples with long chains show smaller fluorescence intensity than samples with shorter chains. This inverse relation indicates that the ligands act as end groups. All these findings can be regarded as a proof that the guanidine ligand is chemically bound to the polymer and remains as end group after polymerisation and work-up.



Scheme 2: Coordination-insertion mechanism for the ROP of lactide with **1** (R = reactants, C = zinc coordinated lactide, TS = transition state, INT = tetrahedral intermediate, P = propagating species)

Until now, only anionic reagents like alkoxides or activated alcohols are reported to promote nucleophilic ring-opening.[1,2] The experimental results support the hypo¬thesis that the neutral, but highly nucleophilic guanidine function can take on this crucial role, too. Herein, we propose the mechanism for the activation and insertion of the first lactide molecule, depicted in Scheme 2, in order to model the initiation step in analogy to the ring-opening reaction with tin ketiminate systems.[1,2] For our theoretical study we chose B3LYP density functional theory (DFT) and the 6-31G(d) and 6 311g+(d) basis sets because previous studies[3] have demonstrated that gas phase DFT calculations using

this combination of functional and basis set are able to describe zinc N-donor complexes reasonably and it has been used successfully for DFT studies on ring-opening polymerisation before.[3] As complex **1** and **2** display similar reactivity,[3b] the remaining discussion focuses on **1**. The difference between these two complexes appeared only in the spectroscopic studies. The resulting reaction coordinate diagram can be found in Figure 2. The starting point of the mechanistic calculations (R) was obtained by substitution of a triflate ion in complex **1** by a lactide molecule. The mechanism of the first lactide insertion starts with the exothermic coordination of the lactide via one of the carbonyl oxygen atoms ($O_{carbonyl)}$ to the zinc centre (C1). In contrast to other, mostly redox active transition metals, Zn^{2+} provides with flexible coordination space and sufficient Lewis acidity. In the transition state TS0 (93 kJ mol^{-1}) the guanidine N atom (N_{gua}) strides away from the zinc centre for a closer coordination of the lactide resulting in C2. During TS1, the N_{gua} atom transfers electron density to the carbonyl C atom ($C_{carbonyl}$) on the activated side of the lactide molecule leading to the formation of a bond (nucleophilic attack). This step needs an activation enthalpy of 102 kJ mol^{-1} which is attainable at elevated temperatures used for these polymerisations. Then, two tetrahedral intermediates INT1 and INT2 are formed exhibiting a very long Zn-N_{gua} distance (3.38 respectively 3.35 Å). In INT2, the second carbonyl oxygen atom participates the zinc coordination (Zn-Oalkoxide 2.28 Å) before TS2 (11 kJ mol^{-1}) is formed where the $C_{carbonyl}$-$O_{alkoxide}$ bond in the lactide molecule breaks, resulting in an eight-membered heterocycle under formation of the ring-opened species O. This heterocycle comprises a Zn-$O_{alkoxide}$ bond (1.85 Å) and a Zn-$O_{carbonyl}$ donor interaction (2.24 Å). Finally, the N-donor atoms of the ligand are released by reorientation of the opened lactide molecule from the zinc and the ligand remains at the end of the new-built chain. This first insertion product P1 contains a five-membered heterocycle of Zn lactate including a Zn-$O_{carbonyl}$ (2.16 Å) interaction.

In general, the calculated reaction proceeds analogously to the mechanisms analysed by Gibson et al.:[7] The transition states TS1 and TS2 are in good accordance in each approach with regard to reaction path and energy profile, but due to the required transformation of the coordination sphere (departure of guanidine ligand and approach of the lactide) the guanidine system exhibits an additional pre-transition state TS0.

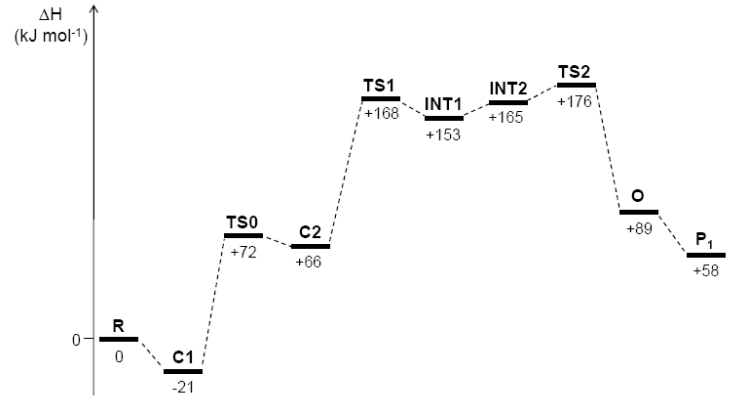
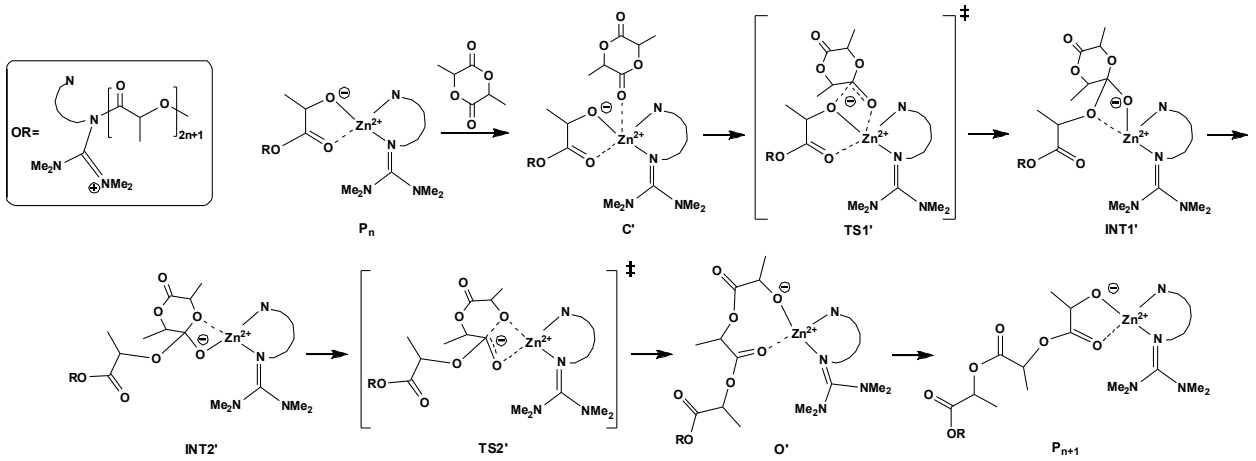


Figure 2: Complete reaction coordinate diagram for the insertion of the first lactide molecule

The chain propagation - insertions of the second and subsequent lactide molecules - has a different character in comparison to the initiation step due to the fact that during the propagation the ring-opening is accomplished by the coordinated alcoholate function of the lactate group (Scheme 3 and Figure 3). Here, a similar reaction coordinate can be derived without the occurrence of the pre-transition state TS0. The characteristic transition states TS1' and TS2' follow the same scheme of nucleophilic attack and subsequent $C_{carbonyl}$-$O_{alkoxide}$ bond release as found for the insertion reaction. Remarkably, the activation barrier



to TS1' is now lowered to 65 kJ mol^{-1} because the alcoholate function of the lactate acts as stronger nucleophile. This value compares well to the experimental value for the activation enthalpy of propagation DH‡ = 79(4) kJ mol^{-1} for the zinc guanidine system **1**. In contrast to the endothermic first insertion step (58 kJ mol^{-1}), the second insertion step proceeds exothermically by 68 kJ mol^{-1} being the driving force of the reaction.

Scheme 3. Coordination-insertion mechanism for the ROP of lactide with P1, model of the propagation step (C = zinc coordinated lactide, TS = transition state, INT = tetrahedral intermediate, P = propagating species)

The propagation step is energetically more favourable than the initiating step. Due to the high initiation barrier a reduced number of active catalyst sites is available for polymerisation in dependence on temperature. This effect explains why the experimental molecular weights are higher than the calculated molecular weights. The higher PD values might also indicate that some catalyst sites start polymerisation later due to the high initiation barrier.

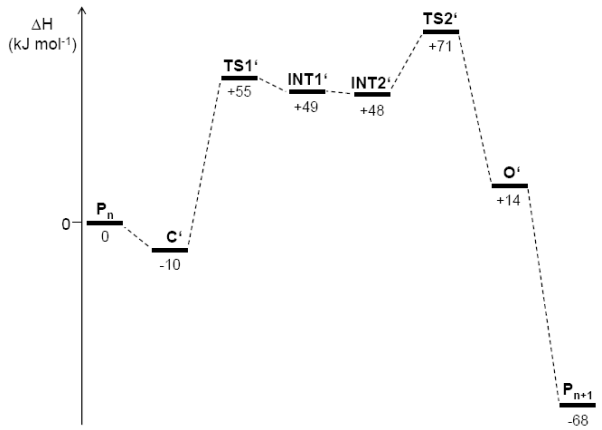


Figure 3: Complete reaction coordinate diagram for the insertion of the second lactide molecule (model for chain propagation)

Conclusion

Experimental and theoretical studies were performed to gain insight into the mechanism of lactide polymerisation mediated by the guanidine-pyridine zinc complexes **1** und **2**. Kinetic studies demonstrated that the reaction obeys first order kinetics and proceeds with living character. UV/Vis and fluorescence measurements indicate that the ligand binds to the polymer and remains as chain end. This together with the absence of racemisation reactions during the polymerisation of pure L-lactide leads to the proposition that the polymerisation proceeds via a coordination-insertion mecha¬nism. The guanidine functions of the coordinated ligand act as nucleophile and accomplish the ring-opening. The role of the only weakly coordinating triflate as non-nucleophile has to be highlighted as it does not coordinationally compete with guanidine or lactide. Hence, these initiators allow the ROP without additional co-catalysts like alcohols or alkoxides. The proposed mechanism was supported by DFT studies. An energy profile of the first lactide insertion step was provided as model for the ROP initiation with 102 kJ mol^{-1} as highest activation barrier. The initiating step shows a reaction coordinate with three characteristic transition states (coordination of lactide, nucleophilic attack and C-O bond release). The propagation step (modelled as insertion of the second lactide) possesses two transition states (nucleophilic attack and C-O bond release) with the highest transition state at 65 kJ mol^{-1} which is in good agreement with the values found experimentally for the chain propagation (79(4) kJ mol^{-1}). These studies are actually extended towards other cyclic esters and they demonstrate that the classical paradigm of anionic ligands for ROP can be overcome.

Resource Usage

All calculations are performed on the ARMINIUS cluster either using Gaussian03 or Turbomole. The submitted jobs are parallel in nature but the efficiency of parallelization

decreases drastically with more than 8 processors. Hence, a mass of sequential jobs (each using 8 processors) are submitted. As quantum chemistry for larger molecules uses a lot of calculation time the immense computing power of the cluster is urgently required for the described project on a daily basis. Besides the processor number, the RAM of 20000 MB per node is highly useful as well for these chemical simulations. Moreover, ARMINIUS offers the possibility of performing long-time-calculations (more than 2 weeks of calculation time). For applications like time-dependent density functional theory, frequency analyses and transition state search, these long times are needed.

References

[1] Platel, R. H.; Hodgson, L.M. and Williams, C.K.: Polym. Rev. 2008, 48, 11-63; b) Gupta, B.; Revagade, N.; Hilborn, J.: Prog. Polym. Sci. 2007, 32, 455-482.

[2] Dechy-Cabaret, O.; Martin-Vaca, B. and Bourissou, D.: Chem. Rev. 2004, 104, 6147-6176; b) Wu, J.; Yu, T.L.; Chen, C.T.; Lin, C.C.: Coord. Chem. Rev. 2006, 250, 602-626.

[3] Börner, J.; Herres-Pawlis, S.; Flörke, U.; Huber, K.: Eur. J. Inorg. Chem. 2007, 5645-5651; b) Börner, J.; Flörke, U.; Huber, K.; Döring, A.; Kuckling, D.; Herres-Pawlis, S.: Chem. Eur. J. 2009, 15, 2362-2376; c) Börner, J.; Flörke, U.; Glöge, T.; Bannenberg, T.; Tamm, M.; Jones, M. D.; Döring, A.; Kuckling, D.; Herres-Pawlis, S.: J. Mol. Cat. A 2010, 316, 139-145.

[4] Wheaton, C. A. and Hayes, P. G.: Dalton Trans. 2010, 3861–3869.

[5] Peckermann, I.; Kapelski, A.; Spaniol, T. P. and Okuda, J.: Inorg. Chem. 2009, 48, 5526-5534.

[6] Hoffmann, A.; Börner, J.; Flörke, U. and Herres-Pawlis, S.: Inorg. Chim. Acta 2009, 362, 1185-1193.

[7] Marshall, E.L.; Gibson, V.C. and Rzepa, H.S.: J. Am. Chem. Soc. 2005, 127, 6048-6051.

7 Summary of References (alphabetical order)

[1] A Science Gateway for Molecular Simulations. Gesing, S., Kacsuk, P., Kozlovszky, M., Birkenheuer, G., Blunk, D., Breuers, S., Brinkmann, A., Fels, G., Grunzke, R., Herres-Pawlis, S., Krüger, J., Packschies, L., Müller-Pfefferkorn, R., Schäfer, P., Steinke, T., Szikszay Fabri, A., Warzecha, K., Wewior, M., and Kohlbacher, O. In: EGI User Forum 2011, Book of Abstracts, pp. 94–95, ISBN 978 90 816927 1 7, April 2011.

[2] Adamo, C. and Barone, V. Journal of Chemical Physics 1999, 110, 6158-6170.

[3] Akenine-Möller, T.; Haines, E. and Homan, N.: Real-Time Rendering 3rd Edition. A. K. Peters, Ltd., Natick, MA, USA, 2008.

[4] Al-Azemi, T.; Kondaventi, L. and Bisht, K.: Macromolecules 2002, 35, 3380.

[5] Amine Bourki et al. Scalability and Parallelization of Monte-Carlo Tree Search. In International Conference on Computers and Games, pages 48-58, 2010.

[6] Argatov, I. and Nazarov, S.: Energy release caused by the kinking of a crack in a plane anisotropic solid. J. Appl. Maths. Mechs. 2002; 66:491-503.

[7] Augustin, W.; Weiss, J.P. and Heuveline, V.: Convey HC-1 hybrid core computer – the potential of FPGAs in numerical simulation. In Prof. Int. Workshop on High-Performance and Hardware-aware Computing, Karlsruhe, Germany, Mar. 2011. KIT Scientific Publishing.

[8] Baker, N.A.; Sept, D.; Holst, M.J. and McCammon, J.A.: The adaptive multilevel finite element solution of the Poisson-Boltzmann equation on massively parallel computers. IBM J. of Research and Development, 45(3.4):427 –438, May 2001.

[9] Bakos, J.: High-performance heterogeneous computing with the convey hc-1. Computing in Science Engineering, 12(6): 80–87, Nov–Dec 2010.

[10] Bangerth, W.; Hartmann, R. and Kanschat, G.: deal. II – a general-purpose object-oriented finite element library. ACM Trans. Math. Softw. 2007; 33(4):4.

[11] Baum, I.; Elsasser, B.; Schwab, L. W.; Loos, K. and Fels, G. Acs Catal 2011, 1, 323.

[12] Beisel, T.; Plessl, C. and Brinkmann, A.: Approaches towards managing heterogeneous Computing Resources in Linux, internal deliverable of the ENHANCE project, September 2011.

[13] Beisel, T.; Wiersema, T.; Plessl, C. and Brinkmann, A.: Cooperative Multitasking for Heterogeneous Accelerators in the Linux Completely Fair Scheduler, in Proceedings of 22nd IEEE International Conference Application-specific Systems Architectures and Processors (ASAP), 2011.

[14] Berente, I.; Beke, T. and Naray-Szabo, G. Theoretical Chemistry Accounts 2007, 118, 129-134.

[15] Binns, F.; Harffey, P.; Roberts, S. M. and Taylor, A.: Journal of Polymer Science Part a-Polymer Chemistry 1998, 36, 2069.
[16] Birkenheuer, G.; Breuers, S.; Brinkmann, A.; Blunk, D.; Gesing, S.; Herres-Pawlis, S.; Krüger, J.; Packschies, L. and Fels, G.: Grid-Workflows in Molecular Science, Proceedings of the Grid Workflow Workshop (GWW), Paderborn, Germany, February 23, 2010.
[17] Brewer, T.: Instruction set innovations for the convey hc-1 computer. Micro, IEEE, 30(2): 70–79, 2010.
[18] Birkenheuer, G.; Blunk, D.; Breuers, S.; Brinkmann, A.; Fels, G.; Gesing, S.; Grunzke, R.; Herres-Pawlis, S.; Kohlbacher, O.; Krüger, J.; Lang, U.; Packschies, L.; Müller-Pfefferkorn, R.; Schäfer, P.; Schuster, J.; Steinke, T.; Warzecha, K. D. and Wewior, M.: MoSGrid: Progress of Workflow driven Chemical Simulations, GWW2011 (in print).
[19] Börner, J.; Herres-Pawlis, S.; Flörke, U.; Huber, K.: Eur. J. Inorg. Chem. 2007, 5645-5651; b) Börner, J.; Flörke, U.; Huber, K.; Döring, A.; Kuckling, D.; Herres-Pawlis, S.: Chem. Eur. J. 2009, 15, 2362-2376; c) Börner, J.; Flörke, U.; Glöge, T.; Bannenberg, T.; Tamm, M.; Jones, M. D.; Döring, A.; Kuckling, D.; Herres-Pawlis, S.: J. Mol. Cat. A 2010, 316, 139-145.
[20] Breslow, R.; Dong, S. D.; Webb, Y. and Xu, R. J. Am. Chem. Soc., 1996, 118, 6588-6600.
[21] Brinkmann, A.; Effert, S., Meyer auf der Heide, F. and Scheideler, C.: „Dynamic and Redundant Data Placement“. In Proceedings of the 27th IEEE International Conference on Distributed Computing Systems (ICDCS), 2007
[22] Brinkmann, A; Effert, S., Gao, Y., Hansen, K. M., Kool, P.: D3.17 Final Storage Architecture Report, Hydra EU Deliverable, Germany, 2009,www.enhance-project.de
[23] Brune, M.; Castaing, L. and Walther, A.: Optimization of Optimal Power Flow Problems. Proceedings in Applied Mathematics and Mechanics 2011, will be published at the end of 2011
[24] Bueckner, H.: A novel principle for the computation of stress intensity factors. ZAMM. 1970; 50:529-546.
[25] CADMEI – Software für Medizinsysteme GmbH, Website: www.cadmei.com
[26] Catalyurek, U. and Aykanat, C.: Hypergraph-partitioning-based decomposition for parallel sparse-matrix vector multiplication. IEEE Transactions on Parallel and Distributed System, 10(7):673–693, 1999.
[27] Chaslot, C.; Winands, M. and Jaap van den Herik, H.: Parallel Monte-Carlo Tree Search. In Conference on Computers and Games, pages 60-71, 2008.
[28] Cheng, H. N. and Maslanka, W. W.; Gu, Q.-M. [US 6677427 2004, Hercules Inc., invs.
[29] Coquet, R.; Hutchings, G.J.; Taylor, S.H.; Willock, D.J. and Mater, J.: Chem., 16 (2006) 1978.

[30] Dawes, A.M.C.; Illing, L.; Clark, S.M. and Gauthier, D.J.: “All-optical switching in Rubidium Vapor”, Science 308, 672 (2005).
[31] Dawes, A.M.C.; Gauthier, D.J.; Schumacher, S.; Kwong, N.H.; Binder, R. and Smirl, A.L.: “Transverse optical patterns for ultra-low-light-level all-optical switching”, Laser & Photonics Reviews 4, 221 (2010).
[32] Dechy-Cabaret, O.; Martin-Vaca, B.; Bourissou, D.: Chem. Rev. 2004, 104, 6147-6176; b) Wu, J.; Yu, T.L.; Chen, C.T.; Lin, C.C.: Coord. Chem. Rev. 2006, 250, 602-626.
[33] Details:http://edgi-project.eu/downloads/-document_library_display/7FkI/view/25605
[34] Dineen, C.; Förstner, J.; Zakharian, A.; Moloney, J. and Koch, S.: Electromagnetic field structure and normal mode coupling in photonic crystal nanocavities. Opt. Express, 13(13): 4980–4985, June 2005.
[35] Donninger, C.; Kure, A. and Lorenz, U.: Parallel Brutus: The First Distributed, FPGA Accelerated Chess Program. In 18th International Parallel and Distributed Processing Symposium. IEEE Computer Society, April 2004.
[36] EDGI Project Website: http://edgi-project.eu/introduction
[37] Elsaesser, B.; Valiev, M. and Weare, J. H. J. Am. Chem. Soc., 2009, 131, 3869-3871.
[38] Elsaesser, B. and Fels, G. J. Mol. Mod. 2011, 17, 1953-1962.
[39] Elsinghorst, P.W.; Cavlar, T.; Müller, A.; Blaut, M.; Braune, A.; Gütschow, M. The thermal and enzymatic taxifolin–alphitonin rearrangement. J. Nat. Prod. 2011, 74, 2243-2249.
[40] Emilsson, G. M.; Nakamura, S.; Roth, A. and Breaker, R. R. RNA-A Publication of the RNA Society 2003, 9, 907-918.
[41] Enzenberger, M. and Müller, M.: A Lock-free Multithreaded Monte-Carlo Tree Search. In 12th International Conference on Advances in Computer Games, volume 6048 of LNCS, pages 14-20. Springer-Verlag, May 2009.
[42] Fox, G.; Williams, R. and Messina, P.: Parallel Computing Works! Morgan Kaufmann, 1994.
[43] Ganguli, A. and Kenig, E. Y.: A CFD-based approach to the interfacial mass transfer at free gas-liquid interfaces. Chem. Eng. Sci. 66 (2011), 3301-3308.
[44] Gao, Y.; Meister, D. and Brinkmann, A.: Reliability Analysis of Declustered-Parity RAID 6 with Disk Scrubbing and Considering Irrecoverable Read Errors,
[45] Gaussian Website, http://www.gaussian.com/g_prod/g09.htm
[46] Gelly, S.: et al. Modification of UCT with Patterns in Monte-Carlo Go. Technical Report 6062, INRIA, 2006.
[47] Granular Security for a Science Gateway in Structural Bioinformatics. Gesing, S., Grunzke, R., Balasko, A., Birkenheuer, G., Blunk, D., Breuers, S., Brinkmann, A., Fels, G., Herres-Pawlis, S., Kacsuk, P., Kozlovszky, M., Krüger, J., Packschies, L., Schäfer, P., Schuller, B., Schuster, J., Steinke, T.,

Szikszay Fabri, A., Wewior, M., Müller-Pfefferkorn, R., and Kohlbacher, O. IWSG-Life 2011 (International Workshop on Science Gateways for Life Sciences), London, UK, June 2011.

[48] Gansel, J.K.; Thiel, M.; Rill, M.S.; Decker, M.; Bade, K.; Saile, V.; von Freymann, G.; Linden, S. and Wegener, M.: „Gold helix photonic metamaterial as broadband circular polarizer“, Science, 325, 1513 (2009).

[49] Gesing, S.; Marton, I.; Birkenheuer, G.; Schuller, B.; Grunzke, R.; Krüger, J.; Breuers, S.; Blunk, D.; Fels, G.; Packschies, L.; Brinkmann, A.; Kohlbacher, O. and Kozlovszky, M.: Workflow Interoperability in a Grid Portal for Molecular Simulations, Proceedings of the International Workshop on Science Gateways (IWSG2010), ed. by R. Barbera, G. Andronico and G. La Rocca, pp. 44-48, Consorzio COMETA ISBN 978-88-95892-03-0.

[50] Gesing, S.; Grunzke, R.; Balasko, A.; Birkenheuer, G.; Blunk, D.; Breuers, S.; Brinkmann, A.; Fels, G.; Herres-Pawlis, S.; Kacsuk, P.; Kozlovszky, M.; Krüger, J.; Packschies, L.; Schäfer, P.; Schuller, B.; Schuster, J.; Steinke, T.; Szikszay Fabri, A.; Wewior, M.; Müller-Pfefferkorn, R. and Kohlbacher, O.: Granular Security for a Science Gateway in Structural Bioinformatics IWSG-Life 2011 (International Workshop on Science Gateways for Life Sciences), London, UK, June 2011 (in print).

[51] Gesing, S.; Kacsuk, P.; Kozlovszky, M.; Birkenheuer, G.; Blunk, D.; Breuers, S.; Brinkmann, A.; Fels, G.; Grunzke, R.; Herres-Pawlis, S.; Krüger, J.; Packschies, L.; Müller-Pfefferkorn, R.; Schäfer, P.; Steinke, T.; Szikszay Fabri, A., Warzecha, K.; Wewior, M. and Kohlbacher, O.: A Science Gateway for Molecular Simulations In: EGI User Forum 2011, Book of Abstracts, pp. 94-95, ISBN 978-90-816927-1-7, 2011.

[52] Giza, M.; Thissen, P. and Grundmeier, G.: Langmuir 24 (2008) 8688.

[53] Glennon, T. M. and Warshel, A. J. Am. Chem. Soc., 1998, 120, 10234-10247.

[54] GRAAP-WG: http://www.ogf.org/gf/group_info/view.php?group=graap-wg

[55] Graf, T.; Lorenz, U.; Platzner, M. and Schaefers, L.: Parallel Monte-Carlo Tree Search for HPC Systems. In Proceedings of the 17th International Conference, Euro-Par 2011, Bordeaux, France, August/September 2011. LNCS, vol. 6853, pp. 365-376. Springer, Heidelberg.

[56] Grawinkel, M.; Pargmann, M.; Dömer, H. and Brinkmann, A.: Lonestar: An Energy-Aware Disk Based Long-Term Archival Storage System, Proceedings of the 17th IEEE International Conference on Parallel and Distributed Systems (ICPADS), Tainan, December, 2011

[57] Grawinkel, M.; Schäfer, T.; Brinkmann, A.; Hagemeyer, J. and Porrmann, M.: Evaluation of Applied Intra-Disk Redundancy Schemes to Improve Single Disk Reliability, Proceedings of the 19th Annual Meeting of the IEEE International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS), Singapore, July, 2011

[58] Grid-Workflows in Molecular Science. Birkenheuer, B.; Breuers, S., Brinkmann, A.; Blunk, D.; Fels, G.; Gesing. S.; Herres-Pawlis, S., Kohlbacher, O.; Krüger, J. and Packschies, L.: Proceedings of the Grid Workflow Workshop (GWW), March 2010

[59] Griffith, A.: The phenomena of rupture and flow in solids. Philos. Trans. Roy. Soc. London 1921; 221:163-198.

[60] Gross, R. A.; Kumar, A. and Kalra, B.: Chemical Reviews 2001, 101, 2097.

[61] Grundmeier, G.; Janke, S.; Ozcan, O. and Birkenheuer, S.: “Surface and thin film engineering of zinc alloy coated steel sheets” Galvatech 2011, 21-24 June 2011

[62] Grunzke, R.; Gesing, S.; Krüger, J.; Birkenheuer, G.; Wewior, M.; Schäfer, P.; Schuller, B.; Schuster, J.; Herres-Pawlis, S.; Breuers, S., Balaskó, A.; Kozlovszky, M.; Szikszay Fabri, A.; Packschies, L.; Kacsuk, P.; Blunk, D.; Steinke, T.; Brinkmann, A.; Fels, G.; Müller-Pfefferkorn, R.; Jäkel, R. and Kohlbacher, O.: A Single Sign-On Infrastructure for Science Gateways on a Use Case for Structural Bioinformatics, Journal of Grid Computing (in print).

[63] Grynberg, G.: “Mirrorless four-wave-mixing oscillation in atomic vapors”, Optics Communications 66, 321 (1988).

[64] Grynko, Y.; Förstner, J.; Meier, T., Radke, A., Gissibl, T.; von Braun, P. and Giessen, H.: Application of the Discontinuous Galerkin Time Domain Method to the Optics of Bi-Chiral Plasmonic Crystals, TaCoNa-Photonics, 2011.

[65] Grynko, Y.; Förstner, J. and Meier, T.: Application of the Discontinuous Galerkin Time Domain Method to the optics of metallic nanostructures, AAAP Vol. 89, Suppl. No. 1, C1V89S1P041 (2011): ELS XIII Conference.

[66] Gu, Q.-M.; Maslanka, W. W. and Cheng, H. N. Polymer Biocatalysis and Biomaterials II, Editor(s): H. N. Cheng, R. A. Gross, ACS Symposium series 2008, 999, Chapter 21.

[67] Harris, M. E.; Dai, Q.; Gu, H.; Kellerman, D. L.; Piccirilli, J. A. and Anderson, V. E. J. Am. Chem. Soc., 2010, 132, 11613-11621.

[68] Hendrickson, B. and Kolda, T.G.: Graph partitioning models for parallel computing. Parallel Comput. 26(12):1519–1534, 2000.

[69] Henkelman, G. and Jonsson, H. Journal of Chemical Physics 2000, 113, 9978-9985.

[70] Hesthaven, J. S. and Warburton, T.: Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Applications. Springer Texts in Applied Mathematics 54. Springer Verlag, New York, 2008.

[71] Himstedt, K.; Lorenz, U. and Möller, D.: A Twofold Distributed Game-Tree Search Approach Using Interconnected Clusters. In Euro-Par, volume 5168 of LNCS, pages 587-598. Springer, 2008.

[72] Hoffmann, A.; Börner, J.; Flörke, U.; Herres-Pawlis, S.; Inorg. Chim. Acta 2009, 362, 1185-1193.

[73] Holtgrewe, M.; Sanders, P. and Schulz, C.: Engineering a scalable high quality graph partitioner. In 22nd International Parallel and Distributed Computing Symposium (IPDPS 2010), pages 1–12. IEEE, 2010.
[74] Huang, M.: VNET: PlanetLab Virtualized Network Access, 2005
[75] IDGF Website: http://desktopgridfederation.org/
[76] Infrastructure Federation Through Virtualized Delegation of Resources and Services. Birkenheuer, G.; Brinkmann, A.; Högqvist, M.; Papaspyrou, A.; Schott, B.; Sommerfeld, D. and Ziegler, W.: In the Journal of Grid Computing, 2011.
[77] ISC2010 http://www.supercomp.de/isc10
[78] Jääskeläinen, S.; Linko, S.; Raaska, T.; Laaksonen, L. and Linko, Y. Y. Journal of Biotechnology 1997, 52, 267.
[79] JSoler, J.M.; Artacho, E.; Gale, J.D.; García, A.; Junquera, J.; Ordejón P. and Sánchez-Portal, D.: Phys.: Condens. Matter, 14 (2002) 2745.
[80] Karger, D.R., Lehman, E.; Leighton, F.T.; Panigrahy, R.; Levine, M.S. and Lewin, D.: “Consistent hashing and random trees: Distributed caching protocols for relieving hot spots on the world wide web”. In Proceedings of the 29th ACM Symposium on Theory of Computing (STOC), 1997, pp. 654–663.
[81] Kasik, D.J.; Buxton, W. and Ferguson, D.R.: Ten CAD challenges. IEEE Computer Graphics and Applications, 25:81-92, March 2005.
[82] Kasik, D.; Dietrich, A., Gobbetti, E.; Marton, F.; Manocha, D.; Slusallek, P.; Stephens, A. and Yoon, S.: Massive model visualization techniques: course notes. In ACM SIGGRAPH 2008 classes, SIGGRAPH '08, pages 40:1-40:188, New York, NY, USA, 2008. ACM.
[83] Keller, M., Meister, D., Brinkmann, A., Terboven, C., & Bischof, C. (2011). eScience Cloud Infrastructure. 2011 37th EUROMICRO Conference on Software Engineering and Advanced Applications (pp. 188-195). Ieee. doi:10.1109/SEAA.2011.38
[84] Keller, M.; Kovacs, J.: A. B. (2011). Desktop Grids Opening up to UNICORE. In M. Romberg, P. Bala, R. Müller-Pfefferkorn, & D. Mallmann (Eds.), UNICORE Summit 2011 Proceedings (pp. 67-76). Torun, Polan: Forschungszentrum Jülich GmbH Zentralbibliothek, Verlag. Retrieved from http://juwel.fz-juelich.de: 8080/dspace/handle/2128/4518
[85] Kenig, E. Y.; Ganguli, A.; Atmakidis, T. and Chasanis, P.: A novel method to capture mass transfer phenomena at free fluid-fluid interfaces. Chem. Eng. Process 50 (2011), 68-76.
[86] Kenter, T.; Plessl, C.; Platzner M. and Kauschke, M.: Estimation and Partitioning for CPU-Accelerator Architectures. Presented at Intel European Research and Innovation Conference (ERIC), October 2011.
[87] Kenter, T.; Plessl, C.; Platzner M. and Kauschke, M.: Performance estimation framework for automated exploration of CPU-accelerator architectures. In

Proc. 19th ACM/SIGDA International Symposium on Field programmable gate arrays (FPGA), pages 177–180. ACM, February 2011.

[88] Kenter, T.; Plessl, C.; Platzner M. and Kauschke, M.: Performance estimation for the exploration of CPU-accelerator architectures. In Omar Hammami and Sandra Larrabee, editors, Proc. Workshop on Architectural Research Prototyping (WARP), Held in conjunction with International Symposium on Computer Architecture (ISCA), June 2010.

[89] Kikuchi, H.; Uyama, H. and Kobayashi, S. Macromolecules 2000, 33, 8971.

[90] Kleinman, L. and Bylander, D.M.: Phys. Rev. Lett., 48 (1982) 1425.

[91] Knani, D.; Gutman, A. L. and Kohn, D. H. J Polym Sci Pol Chem 1993, 31, 1221.

[92] Knuth, E. Donald and Moore, Ronald W.: An Analysis of Alpha-Beta Pruning. In Artificial Intelligence, volume 6, pages 293-327. North-Holland Publishing Company, 1975.

[93] Kobayashi, S.: Macromolecular Rapid Communications 2009, 30, 237.

[94] Kobayashi, S. and Uyama, H.: ACS Symposium series 2003, 840, 128.

[95] Kobayashi, S.; Uyama, H. and Kimura, S. Chem Rev 2001, 101, 3793.

[96] Korst, J. H. M.: “Random duplicated assignment: An alternative to striping in video servers”. In Proceedings of the 5th ACM International Conference on Multimedia (Multimedia), 1997, pp. 219–226.

[97] Krüger, J. and Fels, G.: Ion Permeation Simulations by Gromacs – An Example of High Performance Molecular Dynamics, Concurrency and Computation: Practice and Experience (2010, http://dx.doi.org/10.1002/cpe.1666).

[98] Kück, U.D.; Schlüter, M. and Räbiger, N.: Analyse des grenzschichtnahen Stofftransports an frei aufsteigenden Gasblasen. Chem. Ing. Tech. 81 (2009), 1599-1606.

[99] Kumar, A. and Gross, A.: Biomacromolecules 2000, 1, 133.

[100] Landmann, D.; Plettemeier, D.; Statz, C.; Hoffeins, F.; Markwardt, U.; Nagel, W.; Walther, A.; Herique, A. and Kofman, W.: Three-dimensional reconstruction of comet nucleus by optimal control of Maxwell's equations: A contribution to the experiment CONSERT onboard space mission ROSETTA. Proceedings IEEE International Radar Conference 2010, pp.\,1392-1396 (2010)M.

[101] Langel, W.: Surf. Sci. 496 (2002) 141.

[102] Lazarov, V.K.; Cai, Z.; Yoshida, K., Zhang, K. H. L.; Weinert, M.; Ziemer, K.S. and Hasnip, P.: PRL 107 (2011) 056101

[103] Lensing P.; Meister, D. and Brinkmann, A.: hashFS: Applying Hashing to Optimize File Systems for Small File Reads, Proceedings of the 6th IEEE International Workshop on Storage Network Architecture and Parallel I/Os (SNAPI), Incline Village (NV), May 2010

[104] Lewandowski, P.; Lücke, A. and Schumacher, S.: “All-optical control of transverse polariton patterns in an anisotropic microcavity”, to be presented at the upcoming DPG Spring Meeting, Berlin (2012).
[105] Liferay, http://www.liferay.com/web/guest/partners/sun.
[106] Lim, C. and Tole, P. J. Am. Chem. Soc., 1992, 114, 7245-7252.
[107] Lim, M.S.; Feng, K.; Chen, X.; Wu, N.; Raman, A.; Nightingale, J.; Gawalt, E.S.; Hornak, L.A. and Timperman, A.T.: Langmuir 23 (2007) 2444.
[108] Linux Netfilter, http://www.netfilter.org
[109] Linux-VServer, http://linux-vserver.org
[110] Lischka, J. and Karl, H.: A virtual network mapping algorithm based on subgraph isomorphism detection, VISA '09: Proceedings of the 1st ACM workshop on Virtualized infrastructure systems and architectures, 2009
[111] Louie, S.G., Froyen, S. and Cohen, M.L.: Phys. Rev. B, 26 (1982) 1738.
[112] Luk, M.H.; Tse, Y.C.; Kwong, N.H.; Leung, P.T.; Schumacher, S. and Binder, R.: “Control of transverse optical patterns in semiconductor quantum well microcavities”, to be presented at the upcoming APS March Meeting, Boston (2012).
[113] Marshall, E.L.; Gibson, V.C.; Rzepa, H.S.: J. Am. Chem. Soc. 2005, 127, 6048-6051
[114] Marshall, E.L.; Gibson, V.C. and Rzepa, H.S.: J. Am. Chem. Soc. 2005, 127, 6048-6051.
[115] Maz’ya, V. and Plamenevsky, B.: On the coefficients in asymptotic expressions of the solutions of elliptic boundary-value problems in domains with conical points. Math. Nachr. 1977; 76:29-60.
[116] Mc Evoy, G.V. and Schulze, B.: "Using clouds to address grid limitations," in MGC '08: Proceedings of the 6th international workshop on Middleware for grid computing. New York, NY, USA: ACM, 2008, pp. 1-6. [Online]. Available: http://dx.doi.org/10.1145/1462704.1462715
[117] Meister, D. and Brinkmann, A.: Multi-Level Deduplication in a Backup Scenario, Proceedings of the Israeli Experimental Systems Conference (SYSTOR), Haifa, May, 2009
[118] Meister, D. and Brinkmann; A.: dedupv1: Improving Deduplication Throughput using Solid State Drives (SSD), Proceedings of the 26th IEEE Symposium on Mass Stroage Systems and Technologies (MSST), Incline Village (NV), May 2010
[119] Meyer, B.; Plessl, C. and Förstner, J.: „Transformation of Scientific Algorithms to Parallel Computing Code: Single GPU and MPI multi GPU Backends with Subdomain Support.“ In Symposium on Application Accelerators in High Performance Computing (SAAHPC), 2011. Knoxville, Tennessee, USA, Juli 2011. IEEE.
[120] Meyerhenke, H.: Dynamic load balancing for parallel numerical simulations based on repartitioning with disturbed diffusion. In Proc. Internatl. Conference

on Parallel and Distributed Systems (ICPADS'09), pages 150–157. IEEE Computer Society, 2009.

[121] Meyerhenke, H.: Shape Optimizing Load Balancing for Parallel Adaptive Numerical Simulations Using MPI. Accepted for presentation at 10th DIMACS Implementation Challenge Workshop.

[122] Meyerhenke, H. and Monien, B.: On Multilevel Diffusion-based Load Balancing for Parallel Adaptive Numerical Simulations. Presented at SIAM Conference on Computational Science and Engineering (CSE'11), Reno, (Nevada, USA), February/March 2011.

[123] Meyerhenke, H.; Monien, B. and Sauerwald, T.: A new diffusion-based multilevel algorithm for computing graph partitions. Journal of Parallel and Distributed Computing, 69(9): 750–761, 2009. Best Paper Awards and Panel Summary: IPDPS 2008.

[124] Meyerhenke, H.; Monien, B. and Schamberger, S.: Graph partitioning and disturbed diffusion. Parallel Computing, 35(10–11): 544–569, 2009.

[125] Miranda, A.; Effert, S., Kang, Y., Miller, E.L.; Brinkmann, A and Cortes, T.: "Reliable and randomized data distribution strategies for large scale storage systems". In Proceedings of the 18th International Conference on High Performance Computing (HiPC), 2011

[126] Molecular Simulation Grid. Krüger, J.; Birkenheuer, G.; Breuers, D.; Blunk, S.; Brinkmann, A.; Fels, G.; Gesing, S.; Grunzke, R.; Kohlbacher, O.; Kruber, N.; Lang, U.; Packschies, L.; Herres-Pawlis, R.; Müller-Pfefferkorn, S.; Schäfer, P.; Schmalz, H.G.; Steinke, T.; Warzecha, K.D. and Wewior, M.: German Conference on Chemoinformatics, Goslar, 2010

[127] Molnar, S.; Cox, M.; Ellsworth, D. and Fuchs, H.: A sorting classification of parallel rendering. IEEE Computer Graphics and Applications, 14:23-32, July 1994.

[128] Molnar, S.; Cox, M.; Ellsworth, D. and Fuchs, H.: A sorting classification of parallel rendering. In ACM SIGGRAPH ASIA 2008 courses, SIGGRAPH Asia '08, pages 35:1-35:11, New York, NY, USA, 2008. ACM.

[129] MoSGrid: Progress of Workflow driven Chemical Simulations. Birkenheuer, G., Blunk, D., Breuers, S., Brinkmann, A., Fels, G., Gesing, S., Grunzke, R., Herres-Pawlis, S., Kohlbacher, O., Krüger, J., Lang, U., Packschies, L., Müller-Pfefferkorn, R., Schäfer, P., Schuster, J., Steinke, T., Warzecha, K., and Wewior, M. GWW 2011 (Grid Workflow Workshop), Cologne, Germany, March 2011.

[130] NetNS, http://lxc.sourceforge.net/network.php

[131] Niehörster, O.; Krieger, A.; Simon, J. and Brinkmann, A.: Autonomic Resource Mangement with Support Vector Maschines In: Grid 2011, Proc. of 2011 IEEE/ACM 12th Int. Conf. on Grid Computing, pp. 157–226–128–147164, 2011.

[132] Niehörster, O.; Brinkmann, A.; Fels, G.; Krüger, J. and Simon, J.: Enforcing SLAs in Scientific Clouds. In Proceedings of the 12th IEEE International Conference on Cluster Computing (Cluster2010), Heraklion, 2010.
[133] Niehörster, O.; Brinkmann, A.; Keller, A.; Kleineweber, C.; Krüger, J. and Simon, J.: Cost-Aware and SLO-Fulfilling Software as a Service, (submitted).
[134] Nurmi, D.; Wolski, R.; Grzegorczyk, C.; Obertelli, G.; Soman, S., Youseff, L. and Zagorodnov, D.: "The eucalyptus open-source cloud-computing system," in Proceedings of 9th IEEE International Symposium on Cluster Computing and the Grid, 2009. [Online]. Available: http://open.eucalyptus.com/documents/ccgrid2009.pdf
[135] Oppenheimer, D.; Albrecht, J.; Patterson, D. and Vahdat, A.: Distributed Resource Discovery on PlanetLab with SWORD, In WORLDS, 2004
[136] Park, K. and Pai, V.: CoMon: a mostly-scalable monitoring system for PlanetLab, SIGOPS Oper. Syst. Rev., 2006
[137] Partners: http://edgi-project.eu/partners
[138] Peckermann, I.; Kapelski, A.; Spaniol, T. P.; Okuda, J.: Inorg. Chem. 2009, 48, 5526-5534.
[139] Perdew, J.P.; Burke K. and Ernzerhof, M.: Phys. Rev. Lett., 77 (1996) 3865.
[140] Perreault, D. and Anslyn, E. Angew. Chemie International Edition 1997, 36, 432-450.
[141] Platel, R. H.; Hodgson, L.M. and Williams, C.K.: Polym. Rev. 2008, 48, 11-63; b) Gupta, B.; Revagade, N.; Hilborn, J.: Prog. Polym. Sci. 2007, 32, 455-482.
[142] Popov I.; Brinkmann A. and Friedetzky T.: On the influence of PRNGs on data distribution, Proceedings of 20th Euromicro International Conference on Parallel Distributed and Network-Based Computing (PDP), Garching, Germany, February 2012
[143] Popov, I.; Brinkmann, A. and Friedetzky, T.: “On the Influence of PRNGs on Data Distribution”. In Proceedings of the 20th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP), 2012.
[144] Potente, H. and Thümen, A.: Method for the Optimisation of Screw Elements for Tightly Intermeshing, Co-rotating Twin Screw Extruders, International Polymer Processing (February 2006), pp. 149-154.
[145] Proceedings of the 5th IEEE International Conference on Networking, Architecture and Storage (NAS), Macau, 2010
[146] Raab, M. and Steger, A.: ”Balls into bins” - a simple and tight analysis”. In Proceedings of the Randomization and Approximation Techniques in Computer Science, Second International Workshop (RANDOM), 1998.
[147] Radke, A.; Gissibl, T., Klotzbücher, T.; V. Braun, P. and Giessen, H.: Three-Dimensional Bichiral Plasmonic Crystals Fabricated by Direct Laser Writing and Electroless Silver Plating, Adv. Mater. 23, 3018, (2011).
[148] Raines, R. T. Chemical Reviews 1998, 98, 1045-1065.

[149] Rannacher, R.: Adaptive Galerkin finite element methods for partial differential equations. J. Comput. Appl. Math. 2001; 128:205-233.
[150] Romein, J.W.: et al. Transposition Table Driven Work Scheduling in Distributed Search. In National Conference on Artificial Intelligence, pages 725-731, 1999.
[151] Rubert, S.; Gamroth, C.; Krüger, J. and Sommer, B.: Grid Workflow Approach using the CELLmicrocosmos 2.2 MembraneEditor and UNICORE to commit and monitor GROMACS Jobs, Grid Workflow Workshop GWW2011 (in print).
[152] Rubert, S.; Gamroth, C.; Krüger, J. and Sommer, B.: Managing GROMACS Jobs through Grid Resources using the CELLmicrocosmos 2.2 MembraneEditor IWSG-Life 2011 (International Workshop on Science Gateways for Life Sciences), London, UK, June 2011 (in print).
[153] Scamehorn, C.A.; Harrison, N.M. and McCarthy, M.I.: J. Chem. Phys. 101 (1994) 1547.
[154] Scandolo, S.; Giannozzi, P.; Cavazzoni, C.; de Gironcoli, S., Pasquarello, A. and Baroni, S.: Kristallogr., 220 (2005) 574.
[155] Simon, J.: PC² Benchmarking Center, http://wwwcs.uni-paderborn.de/pc2/about-us/staff/jens-simonspages/benchmarkingcenter.html
[156] Schafer, A.; Horn, H. and Ahlrichs, R. J. Chem. Phys. 1992, 97, 2571.
[157] Schaefers, L.; Platzner, M. and Lorenz, U.: UCT-Treesplit - Parallel MCTS on Distributed Memory. MCTS Workshop, Freiburg, Germany, June 2011.
[158] Schloegel, K.; Karypis, G. and Kumar, V.: Graph partitioning for high performance scientific simulations. In The Sourcebook of Parallel Computing, pages 491–541. Morgan Kaufmann, 2003.
[159] Schumacher, T.; Plessl, C. and Platzner, M.: An Accelerator for k- th Nearest Neighbor Thinning Based on the IMORC Infrastructure. In Proc. Int. Conf. on Field Programmable Logic and Applications (FPL), pages 338–344. IEEE, September 2009.
[160] Schumacher, T.; Plessl, C. and Platzner, M.: “IMORC: Application Mapping, Monitoring and Optimization for High-Performance Reconfigurable Computing,” in Proc. IEEE Symp. on Field-Programmable Custom Computing Machines (FCCM ’09). IEEE, 2009.
[161] Schumacher, T.; Plessl, C. and Platzner, M.: “An Accelerator for k-th Nearest Neighbor Thinning based on the IMORC Infrastructure”, in Proceedings of the 19th International Conference on Field Programmable Logic and Applications (FPL), Prague, Czech Republic, August/September 2009. IEEE
[162] Schumacher, T.; Süß, T.; Plessl, C. and Platzner, M.: “Communication Performance Characterization for Reconfigurable Accelerator Design on the XD1000 ”, in Proc. Int. Conf. on ReConFigurable Computing and FPGAs (RECONFIG '09)
[163] Schumacher, T.; Süß, T.; Plessl, C. and Platzner, M.: FPGA Acceleration of Communication-bound Streaming Applications: Architecture Modeling and a

3D Image Compositing Case Study. International Journal of Reconfigurable Computing (IJRC), vol. 2011, 2011. Article ID 760954.

[164] Schumacher, S.: “Spatial anisotropy of polariton amplification in planar semiconductor microcavities induced by polarization anisotropy”, Physical Review B 77, 073302 (2008).

[165] Schwab, L. W.; Kroon, R.; Schouten, A. J. and Loos, K.: Macromol. Rapid Commun. 2008, 29, 794.

[166] see http://www.intergofed.org/ (website of The International Go Federation) for more information.

[167] Shi, J. and Tomasi, C.: Good features to track, in Proceedings of Computer Vision and Pattern Recognition, IEEE, 1994

[168] Silver, D.: Reinforcement Learning and Simulation-Based Search in Computer Go. PhD thesis, University of Alberta, 2009.

[169] Sommer, B.; Dingersen, T.; Gamroth, C.; Schneider, C.E.; Rubert, S.; Krüger, J. and Dietz, K. J.: CELLmicrocosmos 2.2 MembraneEditor: A modular interactive shape-based software approach to solve heterogenous membrane packing problems, Journal of Chemical Information and Modelling, 51(5):1165-1182, 2009.

[170] Specovius-Neugebauer, M. and Steigemann, M.: Eigenfunctions of the 2-dimensional anisotropic elasticity operator and algebraic equivalent materials. ZAMM. 2008; 88(2): 100-115.

[171] Steigemann, M.: Verallgemeinerte Eigenfunktionen und locale Integral-charakteristiken bei quasi-statischer Rissausbreitung in anisotropen Materialien. Berichte aus der Mathematik, Shaker Verlag: Aachen, 2009.

[172] Steigemann, M.: Simulation of quasi-static crack propagation in functionally graded materials. Functionally Graded Materials, Reynolds N (ed.). Nova Science Publishers, Inc. 2011.

[173] Steigemann, M.: On the precise computation of stress intensity factors and certain integral characteristics in anisotropic inhomogeneous materials. Accepted for publication in Int. J. Numer. Meth. Engng. 2012

[174] Steinbuch Centre for Computing. Karlsruher Institute of Technology. About High Performance Computing as a Service. [Online]. Available: http://www.scc.kit.edu/forschung/4942.php

[175] Süß, T.; Koch, C.; Jähn, C.; Fischer, M. and Meyer auf der Heide, F.: Ein paralleles Out-of-Core Renderingsystem für Standard-Rechnernetze. In Jürgen Gausemeier, Michael Grafe, and Friedhelm Meyer auf der Heide, editors, Augmented & Virtual Reality in der Produktentstehung, volume 295 of HNI-Verlagsschriftenreihe, Paderborn, pages 185-197. Heinz Nixdorf Institut, Universität Paderborn, May 2011.

[176] Süß, T.; Koch, C.; Jähn, C. and Fischer, M.: Approximative occlusion culling using the hull tree. In Proceedings of the Graphics Interface 2011, pages 79-86. Canadian Human-Computer Communications Society, May 2011.

[177] Süß, T.; Wiesemann, T. and Fischer, M.: Evaluation of a c-load-collision-protocol for load-balancing in interactive environments. In Proceedings of the 5th IEEE International Conference on Networking, Architecture, and Storage, pages 448-456. IEEE Computer Society, IEEE Press, July 2010.

[178] Süß, T.; Wiesemann, T. and Fischer, M.: Gewichtetes c-Collision-Protokoll zur Balancierung eines parallelen Out-of-Core-Renderingsystems. In Jürgen Gausemeier and Michael Grafe, editors, Augmented & Virtual Reality in der Produktentstehung, HNI Verlagsschriftenreihe, Paderborn, pages 39-52. Universität Paderborn, HNI Verlagsschriftenreihe, Paderborn, 2010.N. A. Baker, D. Sept, M. J. Holst, and J. A. McCammon. The adaptive multilevel finite element solution of the Poisson-Boltzmann equation on massively parallel computers. IBM J. of Research and Development, 45(3.4):427 –438, May 2001.

[179] Taflove, A. and Hagness, S.: Computational electrodynamics: the finite-difference time-domain method. Artech House antennas and propagation library. Artech House, 2005.

[180] The MoSGrid Gaussian Portlet – Technologies for the Implementation of Portlets for Molecular Simulations. Wewior, M.; Packschies, L.; Blunk, D.; Wickeroth, D.; Warzecha, K.D.; Herres-Pawlis, S.; Gesing, S., Breuers, S.; Krüger, J.; Birkenheuer, G. and Lang, U.: In Proceedings of the International Workshop on Science Gateways (IWSG10), pages 39–43. Consorzio COMETA, 2010.

[181] Thiel, M.; Decker, M.; Deubel, M.; Wegener, M.; Linden, S. and von Freymann, G.: “Polarization stop bands in chiral polymeric three-dimensional photonic crystals “, Adv. Mater. 19, 207 (2007).

[182] Thiel, M.; Rill, M.S.; von Freymann, G. and Wegener, M.: “Three-dimensional bi-chiral photonic crystals“, Adv. Mater, 21, 4680 (2009).

[183] Thurecht, K. J.; Heise, A.; deGeus, M.; Villarroya, S.; Zhou, J. X.; Wyatt, M. F. and Howdle, S. M.: Macromolecules 2006, 39, 7967.

[184] Thissen, P.; Janke, S.; Feil, F.; Fürbeth, W., Tabatabai, D. and Grundmeier, G.: Editor: K.U. Kainer (Hrsg.), Magnesium, Wiley-VCH, Weinheim (2009) 1357.

[185] Tomasi, C. and Kanade, T.: Detection and Tracking of Point Features, in CMU Techreport, 1991

[186] Troullier, N. and Martins, J.L.: Phys. Rev. B, 43 (1991) 1993.

[187] Tse, Y.C.; Luk, M.H.; Kwong, N.H.; Leung, P.T.; Schumacher, S. and Binder, R.: “A low-dimensional mode-competition model for analyzing transverse optical patterns”, in preparation.

[188] Tse, Y.C.; Luk, M.H.; Kwong, N.H.; Leung, P.T.; Schumacher, S. and Binder, R.: ”A low-dimensional population-competition model for analyzing transverse optical patterns”, to be presented at the upcoming APS March Meeting, Boston (2012).

[189] Tusch, M.; Krüger, J. and Fels, G.: Structural Stability of V-Amylose Helices in Water-DMSO Mixtures Analysed by Molecular Dynamics, Journal of Chemical Theory and Computation (2011 dx.doi.org/10.1021/ct2005159, in print).
[190] Turek, S.: Efficient Solvers for Incompressible Flow Problems: An Algorithmic and Computational Approach
[191] Ulman, A.: Chem. Rev. 96 (1996) 1533.
[192] Uyama, H. and Kobayashi, S. Enzyme-Catalyzed Synthesis of Polymers 2006, 194, 133.
[193] Uyama, H. and Kobayashi, S. Chem Lett 1993, 1149.
[194] Valiev, M.; Bylaska, E. J.; Govind, N.; Kowalski, K.; Straatsma, T. P.; Van Dam, H. J. J.; Wang, D.; Nieplocha, J.; Apra, E.; Windus, T. L. and de Jong, W. A.: Computer Physics Communications 2010, 181, 1477.
[195] Valiev, M.; Garrett, B. C.; Tsai, M. K.; Kowalski, K.; Kathmann, S. M.; Schenter, G. K. and Dupuis, M.: J Chem Phys 2007, 127, 51102.
[196] Valiev, M.; Kawai, R.; Adams, J. A. and Weare, J. H.: J Am Chem Soc 2003, 125, 9926.
[197] Valiev, M.; Garrett, B. C.; Tsai, M. K.; Kowalski, K.; Kathmann, S. M.; Schenter, G. K. and Dupuis, M. Journal of Chemical Physics 2007, 127, 51102.
[198] Valiev, M.; Yang, J.; Adams, J. A.; Taylor, S. S. and Weare, J. H. Journal of Physical Chemistry B 2007, 111, 13455-13464.
[199] Valiev, M.; Bylaska, E. J.; Govind, N.; Kowalski, K.; Straatsma, T. P.; Van Dam, H. J. J.; Wang, D.; Nieplocha, J.; Apra, E.; Windus, T. L. and de Jong, W. A. Computer Physics Communications 2010, 181, 1477-1489.
[200] van der Mee, L.; Helmich, F.; de Bruijn, R.; Vekemans, J. A. J. M.; Palmans, A. R. A. and Meijer, E. W.: Macromolecules 2006, 39, 5021.
[201] Varma, I. K.; Albertsson, A.-C.; Rajkhowa, R. and Srivastava, R. K.: Prog. Polym. Sci. 2005, 30, 949.
[202] Wang, J.; Lua, P.; Wang, Z.; Yang, C. and Mao, Z.-S.: Numerical simulation of unsteady mass transfer by the level set method. Chem. Eng. Sci. 63 (2008), 3141-3151.
[203] Wapner, K.; Stratmann, M. and Grundmeier, G.: International Journal of Adhesion & Adhesives 28 (2007) 59.
[204] Westheim, F. Accounts of Chemical Research 1968, 1, 70-&.
[205] Wewior, M.; Packschies, L.; Blunk, D.; Wickeroth, D.; Warzecha, K.; Herres-Pawlis, S.; Gesing, S.; Breuers, S.; Krüger, J.; Birkenheuer, G. and Lang, U.: The MoSGrid Gaussian portlet - Technologies for Implementation of Portlets for Molecular Simulations, Proceedings of the International Workshop on Science Gateways (IWSG2010), ed. by R. Barbera, G. Andronico and G. La Rocca, pp. 39-43, Consorzio COMETA ISBN 978-88-95892-03-0.
[206] Wheaton, C. A. and Hayes, P. G.: Dalton Trans. 2010, 3861–3869.

[207] Wladkowski, B. D.; Krauss, M. and Stevens, W. J. J. Am. Chem. Soc., 1995, 117, 10537-10545.
[208] Workflow Interoperability in a Grid Portal for Molecular Simulations. Gesing, S.; Marton, I.; Birkenheuer, G.; Schuller, B.; Grunzke, R.; Krüger, J.; Breuers, S.; Blunk, D.; Fels, G.; Packschies, L.; Brinkmann, A.; Kohlbacher, O. and Kozlovszky, M.: In Roberto Barbera, Giuseppe Andronico, and Giuseppe La Rocca, editors, Proceedings of the International Workshop on Science Gateways (IWSG10), pages 44–48. Consorzio COMETA, 2010.
[209] WS-Agreement specification: http://www.ogf.org/documents/GFD.107.pdf
[210] WS-PGRADE, https://guse.sztaki.hu/liferay-portal-6.0.5/
[211] Yalagandula, P.; Sharma, P.; Banerjee, S.; Basu, S. and Lee, S.J.: S3: A scalable Sensing Service for Monitoring Large Networked Systems, INM '06: Proceedings of the 2006 SIGCOMM workshop on Internet network management, 2006
[212] Yariv A. and Pepper, D.M.: “Amplified reflection, phase conjugation, and oscillation in degenerate four-wave-mixing”, Optics Letters 1, 16 (1977).
[213] Yee, K. Numerical solution of inital boundary value problems involving maxwell’s equations in isotropic media. IEEE Transactions on Antennas and Propagation, 14:302 – 307, May 1966
[214] Yu, M.; Yi, Y.; Rexford, J. and Chiang, M.: Rethinking Virtual Network Embedding: Substrate Support for Path Splitting and Migration, SIGCOMM Comput. Commun. Rev., 2008
[215] Zegers, I.; Maes, D.; Daothi, M. H.; Poortmans, F.; Palmer, R. and Wyns, L. Protein Science 1994, 3, 2322-2339.
[216] Zentrales Innovationsprogramm Mittelstand, Website: http://www.zim-bmwi.de
[217] Zhu, Z. and Ammar, M.: Overlay network assignment in PlanetLab with NetFinder, Technical Report GT-CSS-06-11, 2006
[218] ZIM Erfolgsbericht http://www.zim-bmwi.de/zim-koop-foerderbeispiele/zim-koop-025.pdf