

Inhalt



Editorial	2
Invasive Computing	4
Iterationsschleife	8
Speech Processing with Java and C++	10
SuperMUC und die vielen Teilchen	11
Neuer Rechencluster CoolMAC	13
CSE bei GAMM und SIAM	15
Vierter BGCE Student Paper Prize	15
Neues aus Copper Mountain	18
SPPEXA News	22
Der Dozent	25
YAPC ²	26
Bitte notieren	27

Das Quartl erhalten Sie online unter <http://www5.in.tum.de/quartl/>



Das Quartl ist das offizielle Mitteilungsblatt des *Kompetenznetzwerks für Technisch-Wissenschaftliches Hoch- und Höchstleistungsrechnen in Bayern* (KONWIHR) und der *Bavarian Graduate School of Computational Engineering* (BGCE)

Editorial

Diesmal wollen wir zunächst schnell beim Bayerischen Rundfunk vorbeischauen, genau gesagt im Web-Bereich von BR-alpha – wer das Folgende genau wissen und alles nachvollziehen will, der mache sich schnell schlau bei

<http://www.br.de/fernsehen/br-alpha/sendungen/alpha-campus/wirtschaft-106.html>

Dort ist ein netter Filmbeitrag zu sehen, in dem man viel über das Fach Betriebswirtschaftslehre lernen kann. Beispielsweise bekennt eine BWL-Studentin freimütig, in der Schule nie ein Mathe-Ass gewesen zu sein (nun gut, beides – also sowohl das Nicht-Sein als auch das Bekennen – scheint ja auch für zahlreiche Promis zu gelten, da ist sie also in bester Gesellschaft) und sich nun dennoch mit Dingen wie Mathematik und Statistik herumschlagen zu müssen. Selbstzweifel plagen die gute Frau allerdings nicht, denn: „Nur weil einem dieses nicht liegt, sollte man nicht vor BWL zurückschrecken“. Das macht uns doch allen gleich Mut im Hinblick auf die nächsten Finanz- und Wirtschaftskrisen. Und dann kommt da auch noch ein gestandener BWL-Professor daher, der im gleichen Kontext mit der frank und frei vorgetragenen Aussage überrascht, das intellektuelle Anforderungsniveau von BWL sei schließlich nicht vergleichbar mit dem der Physik oder (hier platziert der geschätzte Kollege ein überlegtes „vielleicht“) der Ingenieurwissenschaften. Sollte Sie das überraschen, dann gilt der wohl bekannte Matthäus-Spruch „Again what learned“; sollten Sie sich tendenziell eher in Ihren bisherigen vagen Vermutungen bestätigt fühlen (trotz aller interdisziplinären Unvoreingenommenheit, selbstredend), dann machen Sie's einfach wie ich und freuen Sie sich still. Um keine Zweifel aufkommen zu lassen: BWL ist ein tolles Fach. Wirklich.

Dann gilt es noch von einer doch leicht skurrilen Duplizität der Ereignisse zu berichten. Wir schreiben den April des Jahres 2005. Die Kommission für IT-Infrastruktur der DFG, im Volksmund kurz KfR genannt, besucht anderthalb Wochen lang verschiedene Einrichtungen in den USA. Als zweite Station der Reise wird Raleigh angefliegen, Hauptstadt des Bundesstaats North Carolina und eine der Ecken des so genannten „Research Triangle“. Auf dem Programm stehen die North Carolina State University (NCSU) und

IBM. Im Executive Briefing Center von IBM lauschen Gastgeber und Gäste gerade andächtig den Ausführungen mehrerer IBM'ler zum (damals noch brandaktuellen) Cell-Prozessor, als plötzlich eine der Damen vom Empfang aufgeregt hereingestürzt kommt und aus voller Kehle „You are Pope, you are Pope!“ zum besten gibt. Umgehend versammelt sich die ungläubige (etwas doppelsinnig, ich geb's zu) IT-Schar vor einem Großbildschirm und bestaunt den neuen, den deutschen Papst. Dass die legendäre Schlagzeile der BILD-Zeitung am Tag drauf – „Wir sind Papst!“ – somit im Grunde nur die Bestätigung des Ausrufs besagter amerikanischen IBM-Mitarbeiterin war, bleibt wohl für immer eine Episode der Geschichte.

Und nun springen wir fast acht Jahre nach vorne und schreiben den März des Jahres 2013. Erneut bereist die KfR die USA, erneut steht Raleigh auf dem Programm, und erneut gilt unser Besuch der Hauptstadt North Carolinas der NCSU und IBM. Wieder ist in Rom Konklave angesagt, und wieder gibt's ein „Habemus Papam!“ vor Ort, auch wenn diesmal nicht wir, sondern gewissermaßen Maradona und Messi Papst werden. Das führt natürlich zu vergnüglichen Fragen und Äußerungen. „When do you come next time?“, will einer wissen; „I don't know whether I want to see you here again too soon in the future“, meint der Chef der NCSU-Informatik. Ein weiterer Vorschlag lautet, den Tisch beim abendlichen Besuch eines indischen Restaurants gleich für den Tag der nächsten Papstwahl zu reservieren. Etc. etc.

Ergo: Wenn Sie auf dem Laufenden in Sachen Papst bleiben wollen, lassen Sie sich rechtzeitig informieren, wenn die KfR wieder eine USA-Reise plant.

Ziehen wir solche Dinge magisch an? Alte FORTWIHR-Begleiter erinnern sich, dass es etwas Ähnliches schon mal gab. Laut Satzung war ein Posten im FORTWIHR-Beirat einem Vertreter aus der Industrie vorbehalten. Und kaum wurde ein solcher gefunden, überredet und ernannt, da war der Kerl seinen Job los. Also nicht den Beiratsjob, sondern seinen richtigen. Zwei- oder dreimal ist das so passiert. Zu unseren diesbezüglichen „Opfern“ zählt übrigens auch der frühere Bahn-Chef und frisch ernannte Berliner Flughafen-Retter. Für seinen Abgang bei der Bahn können wir aber nix, und für die Zukunft von BER übernehmen wir sowieso keine Verantwortung.

Doch genug gelästert – die ganze Quartl-Redaktion hofft, dass Sie ein

frohes Osterfest genießen konnten und nun bereit für's Frühjahr sind. In diesem Sinne viel Spaß mit der neuen Ausgabe Ihres Quartls und eine gute Zeit!

H.-J. Bungartz

Invasive Computing

Wer jetzt meint, einen Artikel über minimal-invasive medizinische Behandlungsmethoden oder einen Bericht über eine potenzielle Mars-Invasion durch Computer zu lesen, den muss ich leider enttäuschen.

Anlässlich eines kürzlich veröffentlichten Beitrags „The Benefits of Invasive Computing“ im online-Magazin HPC Wire ¹, schildert dieser Quartl-Beitrag den darin erwähnten Forschungsbeitrag mittels „Invasive Computing“ ².

Jeder, der schon einmal eine Applikation auf einem Großrechner gestartet hat, kennt die statischen Parameter, welche hierbei anzugeben sind. Insbesondere die Laufzeit sowie die Anzahl der zu reservierenden Rechen-Ressourcen werden bei jedem Start eines Programms statisch angegeben. Aufgrund dieser Anforderungen wird das Programm solange nicht gestartet, bis entsprechende Ressourcen verfügbar sind.

Aus der Perspektive eines Rechenzentrumsnutzers kann man damit sicher gut leben: Man gibt die Ressourcen und die Zeit, die einem als nützlich erscheinen an, startet die Applikation, und irgendwann bekommt man seine Ergebnisse. Berücksichtigt man nun die für Rechenzentren relevante Zielgröße des Applikationsdurchsatzes, führt diese statische Ressourcenallokation jedoch aus Sicht eines Rechenzentrums zumeist zu einer suboptimalen Nutzung der Ressourcen:

A) Als erstes Beispiel können keine Applikationen gestartet werden, wenn keine zu den Anforderungen passende Ressourcen verfügbar sind. Sollte eine Applikation aufgrund einer bestimmten Gitter-Partitionierung 1024 Rechenressourcen benötigen, aber lediglich 1023 Rechenressourcen frei sind,

¹ http://www.hpcwire.com/hpcwire/2013-02-21/the_week_in_hpci_research.html?page=4

² <http://invasic.informatik.unierlangen.de>

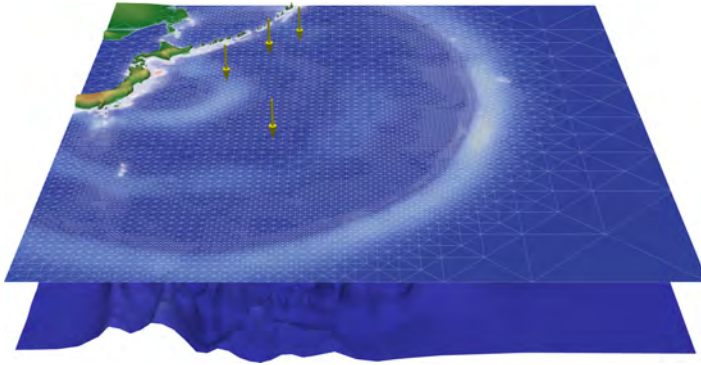


Abbildung 1: Parallele Simulation des Tohoku Tsunami auf einem dynamisch-adaptiven Gitter.

so muss diese Applikation so lange warten, bis eine andere Applikation beendet wurde und die restlichen benötigten Ressourcen zur Verfügung stehen. Solange keine weitere Applikation zum Starten verfügbar ist, welche weniger als 1024 Ressourcen angefordert hat, bleiben diese Rechenressourcen ungenutzt. Als Lösung würde es sich einerseits anbieten, die Applikation lediglich auf 1023 Ressourcen zu starten um die verbleibenden angeforderten Ressource zuzuteilen, sobald diese freigegeben wurde. Eine Alternativlösung wäre, einer Applikation eine Ressource zu entziehen um diese der zu startenden Applikation zuzuweisen.

B) Das zweite Beispiel betrifft science-per-flops Optimierungen mittels dynamisch adaptiver Gitter: Dynamisch adaptive Algorithmen wie z. B. eine parallele dynamisch adaptive Tsunamisimulation³ (siehe Bild 1) optimieren auf die Zielgröße flops-per-science. Eine Implementierung und Ausführung solcher Algorithmen führt zu Applikationen mit einer sich während der Simulation ändernden Rechenlast und damit auch Skalierbarkeit. Aufgrund der statischen Ressourcenverteilung ist jedoch keine Möglichkeit gegeben, Unterschiede der Skalierbarkeit zu berücksichtigen: Einmal zugeteilt, im-

³ <http://www5.in.tum.de/sierpinski/>

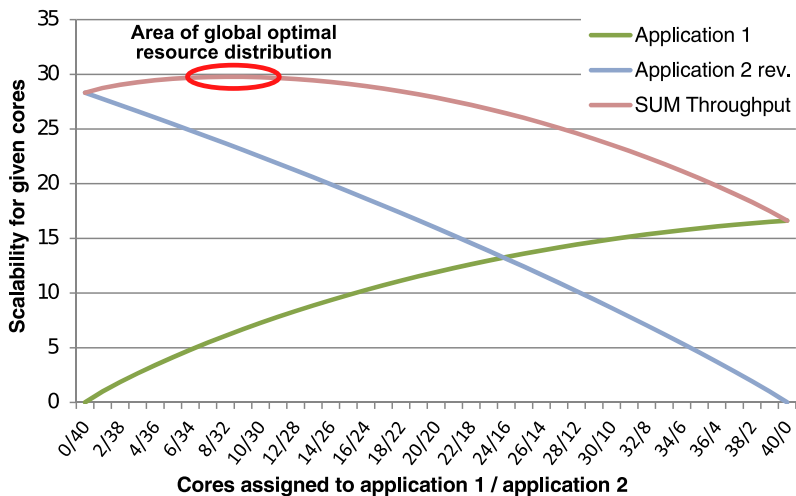


Abbildung 2: Skalierbarkeitsgraphen von zwei Applikationen sowie theoretischer optimaler Gesamtdurchsatz auf einem System mit 40 Rechenressourcen. Die X Achse gibt die möglichen Ressourcenverteilung zu jeder Applikation an.

mer zugeteilt.

Einen Lösungsansatz für die vorgestellten Limitierungen A) und B) bietet nun das Invasive Computing: Das invasive Paradigma schlägt eine dynamische Ressourcenzuteilung während der Programmausführung vor. Ursprünglich für embedded System-on-Chips konzipiert, lässt sich dieses Paradigma auch im HPC-Bereich anwenden. Mittels eines *invades* werden momentane Ressourcenanforderungen wie z. B. die Anzahl der gewünschten Rechenressourcen zur Laufzeit angegeben bzw. modifiziert, ein *infect* führt zu einer Ausführung eines Codes auf der entsprechenden Ressource, und ein *retreat* zur Freigabe der Ressourcen.

Ein zentraler Ressourcenmanager organisiert zur Laufzeit die Zuteilung der Ressourcen mit dem Gesamtdurchsatz als Zielgröße.

Angaben wie z. B. statische Ressourcenanforderungen, bei welchen eine feste Anzahl an Rechenressourcen berücksichtigt werden, stellen für den

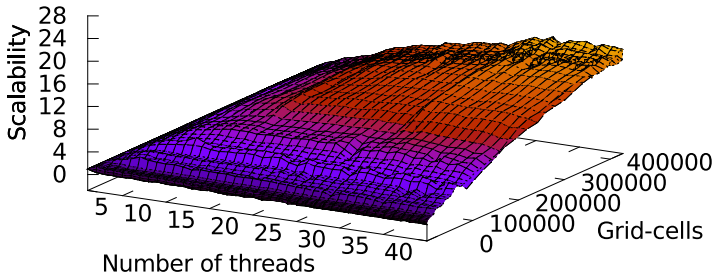


Abbildung 3: Skalierbarkeit einer voll-adaptiven Simulation mit geringer Problemgröße. Bei kleiner Problemgröße liegt eine geringe Skalierbarkeit vor. Eine Applikation in dieser Phase sollte im Vergleich zu einer Applikation, welche Berechnungen auf einem Gitter mit deutlich mehr Last und damit besserer Skalierbarkeit durchführt, weniger Ressourcen zugewiesen bekommen.

Ressourcenmanager lediglich eine applikationsspezifische Sichtweise dar. Aufgrund dieser Informationen könnten lediglich einfache globale Optimierungen durchgeführt werden: Ressourcen, welche von mehreren Applikationen angefordert werden, würden gleichverteilt. Eine globale Optimierung erfordert deshalb neutrale, global auswertbare Angaben, die in Relation zu anderen Applikationen gestellt und für eine Optimierung ausgewertet werden können. Eine solche Angabe ist durch einen Skalierbarkeitsgraphen gegeben. Unter der Annahme der wahrheitsgetreuen Angabe ermöglicht dieser dem Ressourcenmanager eine Optimierung für alle laufenden Applikationen, um die Ressourcenverteilung so durchzuführen, dass ein aufgrund der Skalierbarkeitsgraphen global optimierter Durchsatz erreicht wird (Siehe Bild 2).

Während für Applikationen mit sich nicht ändernden Ressourcenanforderungen ein einziger Skalierbarkeitsgraph ausreichend ist, ist es für Applikationen mit unterschiedlichen Phasen erforderlich, mehrere Skalierbarkeitsgraphen zur Verfügung zu stellen. Ein Skalierbarkeitsgraph für eine voll-adaptive Simulation einer Flachwassergleichung mit sehr geringer Problemgröße und damit relativ geringer Skalierbarkeit ist zur Veranschaulichung in

Bild 3 gegeben. Wie im Skalierbarkeitsgraphen sichtbar, ist für sehr geringe Gittergrößen wie erwartet keine Skalierbarkeit für eine steigende Anzahl an Kernen gegeben. Auch bei 40 Kernen liegt die Skalierbarkeit lediglich bei 1, die Applikation profitiert also nicht von einer steigenden Kernanzahl. Steigt die Anzahl der Gitterzellen, so steigt auch die Skalierbarkeit. Ein um die Problemgröße erweiterter Skalierbarkeitsgraph ermöglicht es dann einer Applikation, dem Ressourcenmanager einen der Gittergröße angepassten Skalierbarkeitsgraphen als Optimierungsinformation zur Verfügung zu stellen.

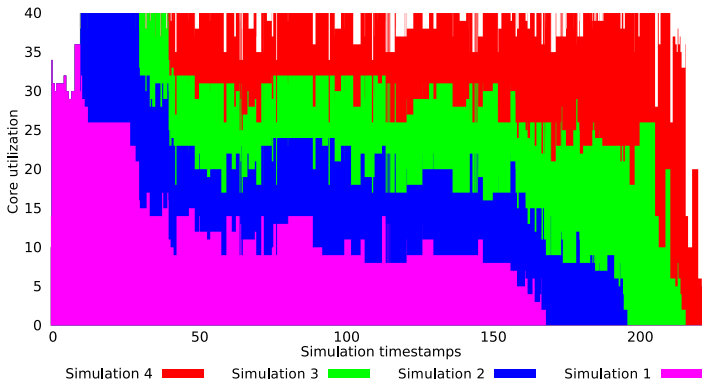


Abbildung 4: Dynamische Zuteilung von Ressourcen für vier zeitversetzt gestartete Shallow-Water Simulationen.

Eine Testimplementierung auf einem 40-Kern shared memory System zeigt, dass sich beim Starten mehrerer invasiver Applikationen mit dynamisch veränderbarer Last eine dynamisch veränderte Ressourcenzuteilung wie im Bild 4 ergibt. Damit sind wir in der Lage Applikationen zu unterschiedlichen Zeitpunkten ohne Ressourcenkonflikte zu starten sowie die sich während der Simulation ändernden Skalierbarkeit durch dynamische Ressourcenumverteilung zu berücksichtigen. Weitere Studien zeigen einen erhöhten Applikationsdurchsatz und damit eine effizientere Ausnutzung der Ressourcen mittels Invasive Computing.

M. Schreiber

Iterationsschleife N=8

11. März 2013

Es war einmal ein Kind das sagte: „Aber er hat ja gar nichts an!“
Damit nähern wir uns dem Ende eines Märchens, das am 7. April 1837 in Dänemark erschienen ist. Geschrieben hat es Hans Christian Andersen (2. April 1805 – 4. August 1875), der noch heute als wichtigster – oder einer der wichtigsten – Schriftsteller Dänemarks gilt. Andersen verfasste eine ganze Sammlung von Märchen, und alle seine Märchen hatten – dem Geist der Zeit geschuldet – eine unschlagbare und offensichtliche Moral und – soweit ich mich erinnern kann – immer ein gutes Ende.

Als ich das Märchen von des Kaisers neuen Kleidern vor rund 40 Jahren las, hielt ich die Erkenntnis des Kindes für einen guten Abschluss. Die Betrüger sind entlarvt, der Kaiser, der Hofstaat und das gesamte Volk sind geheilt, und – so hofft man – sie werden in Zukunft nicht mehr so leicht zu betrügen sein. Die Wahrheit hat über die Dummheit gesiegt.

Heute bin ich skeptischer geworden und frage mich: war es wirklich so? Zunächst ist es völlig unglaubwürdig, dass ein Vater seinem Kind derartige Kritik am höchsten Regierungsvertreter erlaubt. Es ist also davon auszugehen, dass der Vater entweder belehrend (antiautoritär) oder bestrafend (autoritär) weitere regierungskritische Äußerungen des Kindes unterbunden hätte. Wenn jedoch nicht, so wäre davon auszugehen, dass eine derartig irrelevante Bemerkung eines ungebildeten Kindes im besten Fall im allgemeinen Jubel über den Kaiser und seine neuen Kleider untergegangen oder mit einem milden Lächeln abgetan worden wäre. Im schlechteren Fall hätte man das Kind mit Gewalt an weiterer despektierlicher Kritik gehindert. Diese Geschichte kann sich also definitiv nicht so zugetragen haben.

Realistischer ist eher folgendes Ende. Das erstaunte Kind ging nach dem festlichen Umzug nach Hause und begann, sich Gedanken zu machen. Und sehr rasch kam es zu dem Schluss, dass des Kaisers neue Kleider eigentlich wunderbar waren, und dass sich hier eine Marktlücke geöffnet hatte, die es zu füllen galt. Es machte daher zunächst Abitur, schloss ein Wirtschaftsstudium ab – aus Angst vor Plagiatsvorwürfen verzichtete es auf eine Promotion – und stieg dann in die Textilbranche ein.

Heute hat seine Firma Filialen in mehr als 50 Ländern und verkauft dort unter dem Label „Emperor’s“ Konfektion für kleine Leute – auf Wunsch auch in Übergrößen. Produziert wird in Vietnam und Thailand, da dort die Personalkosten niedrig sind – allerdings überlegt das Unternehmen derzeit eine Produktionsverlagerung nach Myanmar. Mit dem Hof versteht sich das früher kritische Kind sehr gut. Es wurde in den Rang eines kaiserlichen Hofkonfektionsrates befördert und dient dem Kaiser immer wieder als Berater in Fragen des Bekleidungsstils. Auf Festen und staatlichen Veranstaltungen ist es gern gesehener Gast.

Einen Teil seines Vermögens hat es vor kurzem einer Stiftung übertragen. Diese soll sich der Erforschung der wissenschaftlichen Grundlagen unsichtbarer Kleider widmen. Erste Ergebnisse sind vielversprechend. Dass die Universität zu Kopenhagen sich mit dem Gedanken an eine Ehrendoktorwürde für den erfolgreichen Unternehmer trage, konnte bisher nicht bestätigt werden.

/co by M. Resch

Run-time Improvements of Speech Processing Algorithms using Java and C++ via the Java Native Interface (JNI)

The Java Speech Toolkit (JSTK)⁴ is developed and maintained by the Speech Group at the University of Erlangen-Nuremberg. It is designed to provide both an API and stand-alone applications for the most popular speech processing tasks such as speech recognition, speaker verification, manual speech transcription and annotation, and evaluation of human rater tasks.

The JSTK is implemented in Java because of several reasons, among them high portability, “developer-friendly” features such as meaningful compile and run-time errors, and the high reproducibility of numeric computations due to the VM implementations. However, using Java comes at the expense of less performance for numerical computations: The VMs, in their current implementation, do not support hardware accelerations such as SIMD or AVX, which can greatly increase the performance of the computations. In a previous project, parts of the JSTK training and evaluation algorithms were parallelized, leading to an approximately linear decrease of the computation time with an increasing number of cores/threads. In this project, the numerical core components are implemented in native code that utilizes both vectorization of simple operations and specialized implementations of the logarithm and exponential function to speed up the computations; the methods are implemented in C++, built on the target machine, and called from Java using the Java Native Interface (JNI).

⁴ <http://code.google.com/p/jstk>

Gaussian mixture models (GMM, a sum of Gaussian probability density functions) are the center piece in most statistical speech processing algorithms such as speech, speaker or speaker state recognition. In a typical scenario, the system faces millions of samples in training, and several thousand for each test utterance (a second of speech typically results in about 100 samples), where the dimension of the sample vectors ranges in between 40 and 90. It is obvious that the main computational effort is to compute the argument to the exponential function (subtract mean from data point, multiply with inverse covariance matrix) and the argument to it.

In total, the number of samples processed per second could be improved by a factor of 10 for diagonal covariance matrices, and about 2.5 for full covariance matrices, by applying the following improvements to the code: Removal of divisions, compiler-friendly indexing, vectorization, loop unrolling, and the restriction to single precision. Further minor improvements could be achieved by replacing the standard implementations of the exponential and log functions by highly optimized versions such as the open source `fmath` or the Intel MKL.

The three main take-home messages are that the basic optimization rules also apply to Java programming, special directives can be used to point the compiler to sections and loops that can should be vectorized or unrolled, and in case of JNI, the invocation overhead needs to be considered.

K. Riedhammer⁵

SuperMUC und die vielen Teilchen: Weltrekord in der Molekulardynamik

Da Anwender aus Ingenieurs- und Naturwissenschaften nach immer längeren und größeren Simulationsläufen verlangen, optimierten wir zusammen mit Kollegen des Leibniz-Rechenzentrums in München sowie von Thermodynamik-Lehrstühlen der TU Kaiserslauten und U Paderborn den Molekulardynamik-Code *ls1 mardyn* für den Intel Sandy Bridge EP Prozessor, das Herzstück des SuperMUC am LRZ.

⁵ korbini.riedhammer@informatik.uni-erlangen.de

Abbildung 1 zeigt ein typisches Anwendungsszenario: Untersuchungen der Tröpfchenbildung von übersättigten Gasen sind technisch von besonderem Interesse, erfordern allerdings große Simulationen mit mindestens mehreren Hunderttausend Molekülen über bis zu einer Million Zeitschritte.

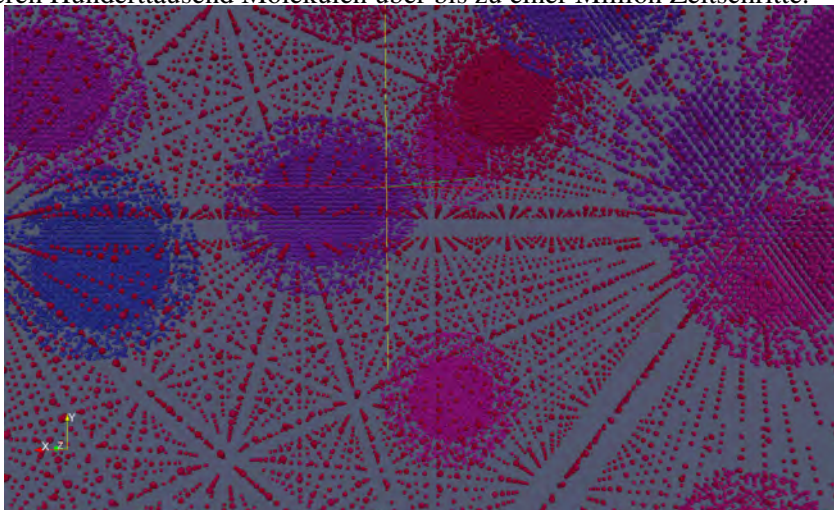


Abbildung 1

Neben der größten Teilchen-Simulation wollten wir natürlich auch überprüfen, ob die Überdruckventile der Warmwasserkühlung vollständig einsatzbereit sind. Dafür haben wir unseren Code “close to Metal” an den neuen Vektorinstruktionssatz (Intel AVX) der SuperMUC Rechenkerne angepasst, wodurch immer bis zu vier Teilchenwechselwirkungen in einem Rutsch berechnet werden können. Zudem haben wir den Speicherbedarf der Anwendung drastisch verkleinert, damit jedes Bit eine (hoffentlich halbwegs) sinnvolle Information speichert: für ein einzelnes Molekül werden nur noch 32 Byte benötigt. Da die Berechnung des Lennard-Jones-12-6-Potentials bösartiger Weise mehr multiplizieren als addieren will, führten wir noch eine leichtgewichtige Shared-Memory-Parallelisierung des Rechenkerns ein. Durch diese Parallelisierung können wir Intel’s Hyper-Threading nutzen und nochmal 12% Performance oben drauflegen, da die Rechenwerke besser genutzt werden können.

Um die Performance bzgl. starker Skalierbarkeit auf SuperMUC zu evaluieren, wurden Läufe mit $N=4.8 \cdot 10^9$ Partikeln ausgeführt. Dabei wurde eine nahezu perfekte Skalierbarkeit auf 146016 Kernen mit 292032 Threads bei einer parallelen Effizienz von 42% von 128 auf 146016 Kerne erreicht.

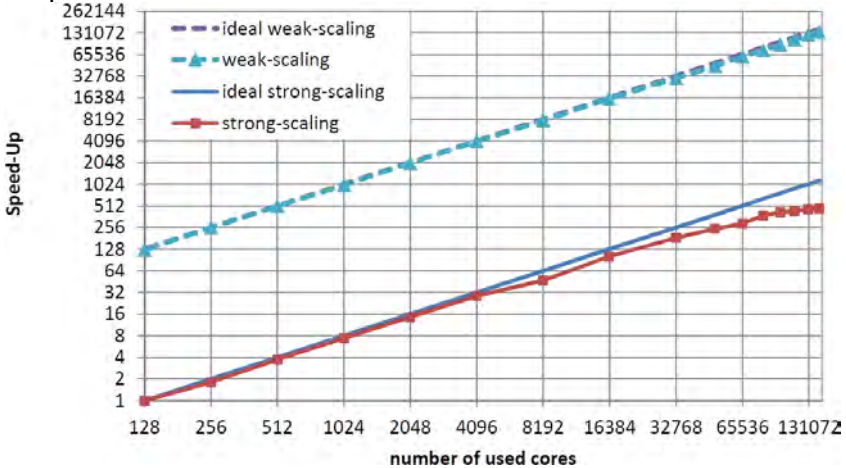


Abbildung 2

Außerdem ließen wir die unseres Wissens nach derzeit größte Molekulardynamik-Simulation mit 4.124.894.592.978 Partikeln auf 146016 Cores laufen, wobei die Berechnung eines Zeitschritts ca. 40 s dauerte. Für dieses Szenario wurde eine absolute Performance von 591.2 TFLOPS mit einem Speedup von 133183 X bezogen auf einen einzelnen Kern erreicht, was einer single-precision Peakperformance von 9.4% entspricht.

W. Eckardt, A. Heinecke

CoolMAC - Neuer Rechencluster für den Munich Centre of Advanced Computing

Seine Lieferung erfolgte zwar noch rechtzeitig zu Weihnachten - die endgültige Inbetriebnahme musste aber noch bis zum Einbruch des Winters warten: seitdem ist "CoolMAC" der neueste Spross der Münchner Cluster-Landschaft. CoolMAC wurde dem Munich Centre of Ad-

vanced Computing (MAC) als Experimentierplattform spendiert – zur Entwicklung neuer Algorithmen für aktuelle Hardwaretechnologien.

Physikalisch ist CoolMAC eine Erweiterung des CoolMUC (Teil des LRZ Linux Clusters), mit dem er die innovative direkte Warmwasserkühlung gemeinsam hat. In den beiden Hauptpartitionen des CoolMAC sind zum Einen 19 Knoten mit jeweils vier AMD Bulldozer Opteron 6274 CPUs (je 2×8 -core; 64 Kerne je Knoten) sowie zum Anderen 28 Knoten mit jeweils zwei Intel SandyBridge-EP Prozessoren (Xeon E5-2670; 16 Kerne je Knoten) verbaut. Dazu gesellen sich drei Partitionen mit jeweils vier Knoten, die für Beschleunigertechnologien reserviert sind (Host-CPU jeweils zwei Intel SandyBridge-EP Xeon E5-2670). Aktuell verrichten dort acht NVIDIA M2090 GPUs sowie ebenfalls acht AMD FirePro W8000 Grafikkarten ihren Dienst. Für die dritte Partition ist eine Ausstattung mit Intel Xeon Phi Co-Prozessoren vorgesehen. Auch das Erforschen Speicher-hungriger Anwendungen soll nicht zu kurz kommen: Hierfür stehen zwei Knoten mit je vier Westmere-EX Xeon E7-4830 und 512GB Hauptspeicher zur Verfügung.

CoolMAC verrichtet mittlerweile tapfer seinen Dienst. Gerüchten zufolge sind bereits einige Prozessoren aufgrund allzu GFlops-hungriger MAC-Anwendungen "abgeraucht". Die Nutzung ist für alle Mitglieder des Munich Centre of Advanced Computing offen. Das Nutzungsmodell des CoolMAC zielt vorrangig auf Experimentier- und Optimierungsläufe ab; für Produktionsläufe sollen weiterhin die Plattformen des LRZ genutzt werden. Interessenten wenden sich bitte an Herrn Alexander Heinecke⁶.

MAC bedankt sich an dieser Stelle ganz herzlich bei allen, die die Beschaffung und Installation des CoolMAC tatkräftig unterstützt haben, seitens des Leibniz Rechenzentrums (v.a. Herbert Huber, Reinhold Bader), der RBG Informatik (Gerhard Hoffmann, Jörg Reiner) sowie des Lehrstuhls Informatik 10 (Josef Weidendorfer, Carsten Trinitis). Besonderer Dank gilt Thomas Auckenthaler und Alexander Heinecke, die diese Last für den MAC maßgeblich getragen haben, sowie (last but not least) der Deutschen Forschungsgemeinschaft und dem Herausgeber des Quartls für die Bereitstellung des nötigen Kleingelds.

M. Bader

⁶heinecke@in.tum.de

CSE bei GAMM und SIAM

Mit mehr als 1300 Teilnehmern hatte die SIAM Conference in Computational Science and Engineering in Boston vom 25.2.-1.3.2013 eine Rekordbeteiligung zu verzeichnen.

Dabei waren die Breite des Themenspektrums - von der mathematischen Modellierung bis hin zum Exa-Scale Computing - und das insgesamt hohe Niveau des Tagungsprogramms beeindruckend. Die Konferenz zeichnete sich außerdem durch eine hohe Beteiligung junger Wissenschaftler und auch eine starke deutsche Präsenz aus. Darüber hinaus wurde auch zum vierten Mal der Studentenpreis der Bavarian Graduate School of Computational Engineering vergeben, über den im Quartl auch noch separat berichtet wird. Das Tagungsprogramm und insbesondere Videoaufzeichnungen der Hauptvorträge können über die URL <http://www.siam.org/meetings/cse13/> aufgerufen werden.

Am Rande der Konferenz in Boston hat auch ein informelles Treffen der Mitglieder des Fachausschusses CSE der GAMM stattgefunden. Themen der Diskussion waren die Entwicklung des Fachs in Deutschland und international, und dabei besonders der aktuell laufende Vorstoß bei der DFG mit dem Ziel einer besseren Berücksichtigung des Fachs CS&E in den DFG-Fachstrukturen. Für die GAMM-Jahrestagung ist es leider nicht gelungen, die Vortragsvorschläge aus dem Fachausschuss CSE zu positionieren.

U. Rüde

Vierter BGCE Student Paper Prize auf der SIAM CS&E 2013

Nach 2007, 2009 und 2011 hat die Bavarian Graduate School of Computational Engineering (BGCE) zum vierten Mal den BGCE Student Paper Prize auf der SIAM Conference on Computational Science and Engineering verliehen. Doktoranden und Studierende aus aller Welt nahe ihres Abschlusses konnten ihre Forschungsarbeit in Form eines vierseitigen Kurzpapiers einreichen. Die besten acht durften als Finalisten für den Preis ihre Beiträge auf der SIAM CS&E 2013 in Boston, Massachusetts, präsentieren.

Der BGCE-Preis bleibt eine Erfolgsgeschichte: Die hohe Anzahl an Einreichungen von 2011 konnte gehalten werden: 20 Beiträge von 18 verschiedenen Forschungseinrichtungen und Universitäten aus 6 Ländern (darunter viele anerkannte Adressen und Arbeitsgruppen und natürlich unter Ausschluss der auslobenden Universitäten FAU und TUM) gingen zur Vorauswahl ein. Neben der reinen Menge war auch diesmal wieder die Qualität sehr hoch. Aus diesem interessanten Pool wurden folgende acht Finalbeiträge ausgewählt:

Sergio Amaral (MIT):	A decomposition approach to uncertainty analysis of multidisciplinary systems
Jun Fang (Chinese Academy of Sciences):	A symmetry-based decomposition approach to Eigenvalue problems with applications to electronic structure calculations
Erich Foster (Virginia Tech):	Conforming finite elements for the pure streamfunction form of the quasi-geostrophic equations of the ocean
Tobin Isaac (University of Texas):	Advanced simulation of polar ice sheets: meshing, parallel adaptivity, high-order discretization, robust scalable solvers, and inversion for basal boundary conditions
Lorenz John (TU Graz):	Optimal control problems in energy spaces and applications to arterial blood flow
Jonas Šukys (ETH Zürich):	Multi-level Monte Carlo finite volume methods for stochastic systems of hyperbolic conservation laws
Noud Werter (TU Delft):	An advanced aeroelastic tailoring tool for wing optimisation
Minghao Wu (University of Maryland):	Lyapunov inverse iteration for computing the rightmost eigenvalue of large generalized eigenvalue problems

Eine internationale Jury bewertete die Beiträge (herzlichen Dank auch an dieser Stelle nochmals an Weizhu Bao, Lois McInnes, Scott MacLachlan, Pete Turner und Barbara Wohlmuth; vervollständigt wurde die Jury durch die Ausrichter Hans-Joachim Bungartz und Ulrich Rüde).

In einem Kopf-an-Kopf-Rennen setzte sich schließlich Tobin Isaac von der University of Texas in Austin durch. Der Vollständigkeit halber sei erwähnt, dass die Länge des Beitragstitels dabei kein Bewertungskriterium darstellte.



BGCE-Preis-Finalisten und Jury-Mitglieder (v.l.n.r.): Lorenz John, Hans-Joachim Bungartz, Pete Turner, Barbara Wohlmuth, Lois McInnes, Ulrich Rüde, Jun Fang, Scott MacLachlan, Minghao Wu, Jonas Šukys, Noud Werter, Preisträger Tobin Isaac, Sergio Amaral und Weizhu Bao

Der Preis besteht aus einem einwöchigen Besuch an der TUM und der FAU Erlangen, im Rahmen dessen der Preisträger sowohl Gelegenheit haben wird, die beiden Universitäten kennenzulernen, als auch, seine eigene Forschungsarbeit in jeweils einem Vortrag im Detail vorzustellen. Wir freuen uns sehr auf den anstehenden Besuch von Tobin Isaac mit interessanten Diskussionen im Sommer diesen Jahres.

T. Neckel

Neues aus Copper Mountain

zum 30. Jubiläum der Copper Mountain Conference on Multigrid Methods (CMCMM)

Die Anreise nach Copper Mountain (Ski Resort in Colorado), das vom 17.03. bis zum 22.03.2013 das 14. Mal Austragungsort der CMCMM war⁷, gestaltete sich für unser fünfköpfiges Team aus Erlangen alles andere als angenehm.

Harald Köstler hatte mit zwei Stunden noch die geringste Verspätung, während Ulrich Rüde (Opfer eines Schneesturms in Frankfurt), Björn Gmeiner (aufgehalten von der US Einreisebehörde), Kai Hertel und Daniel Ritter (beide ausgebremst durch ein Metallteil auf der Interstate) mit sechs bis zwölf Stunden Verzögerung am Ziel ankamen. Beim Betreten unseres reservierten Apartments bemerkten wir eine heimelige Unaufgeräumtheit (Essensreste und schmutziges Geschirr in der Küche, fehlende Handtücher, benutzte Betten). Lebten hier etwa noch weitere Personen? Zum Glück handelte es sich nur um ein Versehen, das schnell korrigiert werden konnte.

Die Konferenz selbst war glücklicherweise besser organisiert: Es wurde mehr gegittert (geometrisch, algebraisch, auf Graphen und irgendwo zwischen Räumen und in Zwischenräumen) als jemals zuvor. Zum 30. Geburtstag der Konferenz wurde auch ein Schrein mit heiligen Artefakten („sacred papers“) aus den Anfangsjahren der Konferenz präsentiert, z.B. die ersten Ausgaben des (mit der Schreibmaschine verfassten) „Multigrid Newsletters“, den Steve McCormick seit 1981 herausgab. Einige der Materialien sind inzwischen auch online und somit der Öffentlichkeit nicht mehr vorzuenthalten⁸. Es lohnt sich durchaus, dort mal hineinzuschauen: Van Emden Henson wies beim Konferenz-Dinner darauf hin, dass sich in der zweiten Ausgabe des „Multigrid Newsletter“ – neben skeptischen Zuschriften, die an der neuen Methode zweifelten – ein Aufruf des Herausgebers befindet, der um Zuschriften bittet, die die allgemeine Bedeutung der Mehrgitterverfahren über einfache Anwendungen hinaus belegen sollen. Fragen, die auch

⁷Ja, es ist die 15. CMCMM, allerdings tatsächlich erst die 14. in Copper Mountain, da sie einmal anderenorts stattfand.

⁸ http://grandmaster.colorado.edu/copper/Copper_1st/



Abbildung 5: Ausbremsendes Metallteil aus dem Antriebsstrang eines LKW

heute nicht vollständig beantwortet sind und mit denen sich jeder „Multi-gridder“ auseinandersetzen sollte.

Nach Rückbesinnung auf das Alte nun zu den wesentlichen Neuerungen der Konferenz. Hiervon gab es zwei: Die erste ist ein neuer Gedichttypus: Das Limer-ku, entstanden als Crossover aus Limerick und Haiku. Hierbei werden die Regeln (Versmaß, Endreim, und inhaltliche Referenzen) aus beiden Elter-Gedichttypen derart durcheinander geschüttelt und ad absurdum geführt, dass kaum noch ein sinnvolles gültiges Gedicht geschrieben werden kann. Ich fand jedenfalls keines, obwohl ein verlockender Preis winkte: eine Flasche Riesling, die zuvor über 5000 Meilen in Ulrich Rüdes Skischuhen verbracht hatte.

Tatsächlich sind die Regeln des Limer-kus so streng, dass sogar die Gewinner nur ein Gedicht zustande brachten, das eine Kombination aus einer

Persiflage auf das Versmaß in den ungeraden Zeilen (tah-tatata-tah-tah o.ä.) und einer Schmähung des Vorsitzenden ob der Länge seiner Rede darstellte. Kein besonders wertvoller poetischer Beitrag!



Abbildung 6: Mehrgitter Showcase

Die zweite – und bei Weitem interessantere – Neuerung war der auf Vorschlag von Irad Yavneh von Harald Köstler organisierte „Multigrid Showcase“, ebenfalls ein Crossover, aber diesmal aus Poster-Session und Hands-On-Workshop. Der Showcase fand am Dienstag Abend statt, und schon die Teilnehmerzahl war im Vergleich zu früheren derartigen Sonderveranstaltungen erfreulich hoch. Insgesamt hatte Harald sieben Teams motivieren können, ihre Mehrgitter-Frameworks, Visualisierungstools oder Bildverarbeitungsprogramme live zu präsentieren und die interessierten Besucher damit herumexperimentieren zu lassen. Das Feedback war überwiegend positiv, wobei die einzige Kritik war, dass noch mehr Personen gerne



Abbildung 7: Eingeschneites Flugzeug

ihre Software präsentiert hätten, aber zu kurzfristig vom Showcase erfahren hatten. Daher wird das Ganze hoffentlich in den kommenden Jahren in einem größeren Rahmen wiederholt werden. Ich finde, dass diese Veranstaltung eine perfekte Ergänzung zu den häufig theoretischen Vorträgen auf der Konferenz ist. Hier wird schön illustriert, welche tollen Programme von der Mehrgitter-Gemeinde entwickelt wurden (und werden). Erlangen war übrigens mit zwei Beiträgen zum SPPEXA-Projekt Terra-NEO und zum Visualisierungs-Framework VIPER prominent vertreten.

Die Zeit in Copper Mountain verging insgesamt wieder einmal viel zu schnell (dank des Austausches mit anderen Wissenschaftlern und der angenehmen Wintersportmöglichkeiten in den Pausen). Trotz eines über Colorado hereinbrechenden Blizzards, war die Heimreise wesentlich pünktlicher und weitaus weniger chaotisch als die Anreise. Der Anblick des verschnei-

ten Flugzeuges verhiß zwar nichts Gutes, aber nach einer gut einstündigen Enteisung waren wir dann trotzdem startklar. Einige Kollegen kamen sogar früher an als erhofft, so dass sie etwas für die verspätete Ankunft entschädigt wurden.

D. Ritter

SPPEXA News

Das erste SPPEXA Annual Plenary Meeting ist abgeschlossen, und schon stehen weitere Veranstaltungen wie SPPEXA @ ISC2013 am Programm.



Einmal jährlich versammelt sich die gesamte SPPEXA Community um sich über aktuelle Fortschritte und mögliche Kooperationen auszutauschen. Das Meeting bietet auch ein Forum, um über strategische Fragen des Schwerpunktprogramms zu diskutieren. Am 22. März fand nun das erste dieser SPPEXA Annual Plenary Meetings an der Technischen Universität München statt. Da SPPEXA erst am 1. Januar 2013 vollständig startete, war das Treffen diesmal noch etwas verkürzt und dauerte nur einen Tag. Das änderte nichts daran, dass alle 13 Projekte vertreten waren. Insgesamt nahmen mehr als 60 Personen teil. Die Veranstaltung war aufgeteilt in einen organisatorischen Teil und in einen fachlichen Teil.

Im organisatorischen Teil gaben die zwei Koordinatoren des Programms einen Überblick über den Aufbau von SPPEXA, klärten viele Fragen zu den formalen Abläufen und stellten das Konzept zum Verteilen der Verbundmittel vor. Da sich das Format von SPPEXA stark an das eines SFBs anlehnt, wurde eine Satzung erstellt und auf der Versammlung beschlossen. In der Satzung werden die einzelnen Gremien, Versammlungen und Wahlen festgelegt.

Die erste Wahl wurde anschließend auch sofort durchgeführt. Das Steering Committee wurde offiziell von den SPPEXA Mitgliedern gewählt. Laut SPPEXA Satzung zählt zu den Aufgaben des Steering Committees, dass es die erforderlichen SPPEXA-weiten Maßnahmen für das Erreichen der Ziele von SPPEXA durchführt. Dies umfasst insbesondere die Verteilung

der Verbundmittel. Neben den zwei Koordinatoren (Hans-Joachim Bungartz (TUM), Wolfgang E. Nagel (TU Dresden)), die immer dem Steering Committee angehören, wurden Hans-Peter Bunge (LMU), Christian Lengauer (U Passau), Sabine Roller (U Siegen), und Dörte Sternel (TU Darmstadt) gewählt.



Jedes SPPEXA Projekt stellte sich in 12 Minuten kurz vor.

Im fachlichen Teil stellten sich die 13 Projekte in einem kurzen 12-minütigen Vortrag vor. Das Ziel war allen TeilnehmerInnen einen Überblick über die Projekte in SPPEXA zu geben und etwaige Kooperationsmöglichkeiten zu eruieren. Obwohl die 12 Minuten etwas knapp waren, wurden sie überraschend präzise eingehalten und die Pausen für weitere Diskussionen genutzt. Bereits in den Vorträgen nach dem Mittagessen gab es erste konkrete Kooperationsvorschläge.

Am Ende wurde noch auf zukünftige SPPEXA Veranstaltungen hingewiesen. Es wird im Herbst 2013 das erste Doctoral Retreat & Coding Week geben, wo sich DoktorandInnen aus den SPPEXA Projekten versammeln, die Sichtweisen der anderen kennenlernen, und gemeinsam die Koopera-



Das Mittagessen stärkte nicht nur die TeilnehmerInnen sondern auch die Kooperationen zwischen den Projekten.

tionen zwischen den Projekten vorantreiben. Hierfür soll es auch explizit “hands-on sessions” geben.

Davor, bereits im Juni 2013, gibt es eine SPPEXA Veranstaltung auf der ISC 2013 in Leipzig. SPPEXA konnte sich einen Konferenzraum am Mittwoch, 19. Juni 2013 sichern. Neben Vorträgen mit ersten Ergebnissen aus den Projekten werden auch internationale Gäste über HPC & Co sprechen. Mehrere Pausen mit entsprechender Verpflegungen bieten einen hervorragenden Rahmen um über die SPPEXA Projekte im allgemeinen und die Vorträge im besonderen zu diskutieren. SPPEXA @ ISC2013 wird mit einer Podiumsdiskussion über die Grenzen von Simulation und HPC schließen – hoffentlich nicht mit zu trüben Aussichten für den weiteren Verlauf des Schwerpunktprogramms.

B. Peherstorfer

Der Dozent

Voll Würde wie ein Präsident
und als sein eig'nes Monument
steht vor dem Whiteboard der Dozent,
das Podium ist sein Postament.

Er spricht teils leis', teils vehement,
teils ruhig und teils „mit die Händ“,
teils mit, teils ohne Temp'rament
oft auch mit bay'rischem Akzent,
zuweilen sogar mit Talent
und in der Regel permanent
zu einem Publikum, das pennt.

Das Rechnen ist sein Element
in ihm fühlt er sich kompetent
und manchmal zeigt er auch dezent,
wie klug er ist und wie patent.

Dann schweift sein Blick zum Firmament,
sein Lächeln wirbt für Pepsodent
(beziehungsweise Kukident)
und jeder spürt es: Sapperment!
Heut' ist er wieder exzellent!
In solch erhebendem Moment
erstarrt vor Kummer der Student;
in seiner Brust die Frage brennt:
Sind die Probleme so horrend?
Bin ich so unintelligent?
Liegt mein Intelligenzquotient
wahrhaftig nur bei 10 Prozent?

Doch irgendwann er dann erkennt:
Ein jedes Wissen bleibt Fragment!

Ein bisschen weiß auch der Student,
ein bisschen mehr der Absolvent
unglaublich viel der Assistent
und auch nicht alles:
der Dozent!

D. Weber

Save the Date

Dem Lehrstuhl für Informatik 5 ist es in Kooperation mit dem Leibniz-Rechenzentrum gelungen, den Zuschlag für die YAPC² 2014 zu bekommen. Die Konferenz wird am 21. – 25. Juli 2014 am IAS der TUM stattfinden.

Es bietet sich also an, die entsprechenden Tage bereits jetzt zu blocken. Unten die englische Ausschreibung, die in Kürze über NADigest& Co verschickt werden wird – Quartl hatte sie zuerst!

The International 2014 YAPC² will take place at the TUM Institute of Advanced Study, 21-25 July 2014. The conference comprises invited, organized and contributed sessions as well as a couple of tutorials.

The conference aims at bringing together researchers and practitioners to discuss recent developments in parallel and high-performance computing. It is a joint event with the PPAM⁷14, EuroPar, EuroMPI, ParCo, ISPDC, and Para conferences and covers all topics already covered there. Conference highlights are

- a special session on GPU programming called „Standard relaxation methods on regular Cartesian grids with a speedup > 10000“,
- a daily keynote by Jack Dongarra, and
- an industry workshop by a major hardware vendor with the title „First experiences with a hardware that is not yet released, totally different than everything else – in particular graphics cards, that hardly anybody will be able to use, but that will change everything.“

The official name of the latter event will be revealed as soon as participants sign an NDA.

Conference proceedings will be printed twice: All papers first are published by Elsevier, but as everybody boycotts Elsevier, we will also publish them in LNCS as volume 212456.2135.b.2. We call for contributed papers in English Language. Extended abstracts of at least two pages should be submitted in electronic form by 1 April 2014. Proposals are to be submitted in .pdf-format. Full papers are due before the conference.

On behalf of the YAPC² (Yet Another Parallel Computing Conference),
April 1, 2013

T. Weinzierl

Bitte notieren

- Die Ferienakademie wird 30: Dieses Jahr zieht die Ferienakademie zum 30. Mal in's Sarntal (22. September bis 4. Oktober); wie immer gibt es ein reichhaltiges Kursprogramm für begabte und interessierte Studierende der ausrichtenden Universitäten – insbesondere auch aus den Studiengängen im Quartl-Umfeld.

Näheres dazu und das Online-Bewerbungssystem (Bewerbungsschluss ist der 12. Mai) unter

<http://www.ferienakademie.de/>

Quartl* - Impressum

Herausgeber:

Prof. Dr. A. Bode, Prof. Dr. H.-J. Bungartz, Prof. Dr. U. Rüde

Redaktion:

J. Daniel, C. Halfar, B. Peherstorfer, Dr. S. Zimmer

Technische Universität München, Fakultät für Informatik

Boltzmannstr. 3, 85748 Garching b. München

Tel./Fax: ++49-89-289 18630 / 18607

e-mail: halfar@in.tum.de, **www:** <http://www5.in.tum.de/quartl>

Redaktionsschluss für die nächste Ausgabe: **30.7.2013**

* **Quartel**: früheres bayerisches Flüssigkeitsmaß,

→ das **Quart**: $1/4$ Kanne = 0.27 l

(Brockhaus Enzyklopädie 1972)