

# Tools for High Performance Computing 2015

Andreas Knüpfer · Tobias Hilbrich  
Christoph Niethammer · José Gracia  
Wolfgang E. Nagel · Michael M. Resch  
Editors

# Tools for High Performance Computing 2015

Proceedings of the 9th International  
Workshop on Parallel Tools  
for High Performance Computing,  
September 2015, Dresden, Germany

 Springer

*Editors*

Andreas Knüpfer  
Zentrum für Informationsdienste und  
Hochleistungsrechnen (ZIH)  
Technische Universität Dresden  
Dresden  
Germany

Tobias Hilbrich  
Zentrum für Informationsdienste und  
Hochleistungsrechnen (ZIH)  
Technische Universität Dresden  
Dresden  
Germany

Christoph Niethammer  
Höchstleistungszentrum Stuttgart (HLRS)  
Universität Stuttgart  
Stuttgart  
Germany

José Gracia  
Höchstleistungszentrum Stuttgart (HLRS)  
Universität Stuttgart  
Stuttgart  
Germany

Wolfgang E. Nagel  
Zentrum für Informationsdienste und  
Hochleistungsrechnen (ZIH)  
Technische Universität Dresden  
Dresden  
Germany

Michael M. Resch  
Höchstleistungszentrum Stuttgart (HLRS)  
Universität Stuttgart  
Stuttgart  
Germany

---

*Cover front figure:* OpenFOAM Large Eddy Simulations of dimethyl ether combustion with growing resolutions of 1.3 million elements, 10 million elements, and 100 million elements from left to right reveal how more computing power produces more realistic results. Courtesy of Sebastian Popp, Prof. Christian Hasse, TU Bergakademie Freiberg, Germany.

---

ISBN 978-3-319-39588-3

ISBN 978-3-319-39589-0 (eBook)

DOI 10.1007/978-3-319-39589-0

Library of Congress Control Number: 2016941316

Mathematics Subject Classification (2010): 68U20

© Springer International Publishing Switzerland 2016

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made.

Printed on acid-free paper

This Springer imprint is published by Springer Nature

The registered company is Springer International Publishing AG Switzerland

# Preface

Highest-scale parallel computing remains a challenging task that offers huge potentials and benefits for science and society. At the same time, it requires deep understanding of the computational matters and specialized software in order to use it effectively and efficiently.

Maybe the most prominent challenge nowadays, on the hardware side, is heterogeneity in High Performance Computing (HPC) architectures. This inflicts challenges on the software side. First, it adds complexity for parallel programming, because one parallelization model is not enough; rather two or three need to be combined. And second, portability and especially performance portability are at risk. Developers need to decide which architectures they want to support. Development or effort decisions can exclude certain architectures. Also, developers need to consider specific performance tuning for their target hardware architecture, which may cause performance penalties on others. Yet, avoiding architecture specific optimizations altogether is also a performance loss, compared to a single specific optimization. As the last resort, one can maintain a set of specific variants of the same code. This is unsatisfactory in terms of software development and it multiplies the necessary effort for testing, debugging, performance analysis, tuning, etc. Other challenges in HPC remain relevant such as reliability, energy efficiency, or reproducibility.

Dedicated software tools are still important parts of the HPC software landscape to relieve or solve today's challenges. Even though a *tool* is by definition not a part of an application, but rather a supplemental piece of software, it can make a fundamental difference during the development of an application. This starts with a debugger that makes it possible (or just more convenient and quicker) to detect a critical mistake. And it goes all the way to performance analysis tools that help to speed up or scale up the application, potentially resolving system effects that could not be understood without the tool. Software tools in HPC face their own challenges. In addition to the general challenges mentioned above there is the *bootstrap challenge*—tools should be there early when a new hardware architecture is

introduced or an unprecedented scalability level is reached. Yet, there are no tools to help the tools to get there.

Since the previous workshop in this series, there have been interesting developments for stable and reliable tools as well as tool frameworks. Also there are new approaches and experimental tools that are still under research. Both kinds are very valuable for a software ecosystem, of course. In addition, there are greatly appreciated verification activities for existing tools components. And there are valuable standardization efforts for tools interfaces in parallel programming abstractions.

The 9th International Parallel Tools Workshop in Dresden in September 2015 included all those topics. In addition, there was a special session about user experiences with tools including a panel discussion. And as an outreach to another community of computation intensive science there was a session about Big Data algorithms. The contributions presented there are interesting in two ways. First as target applications for HPC tools. And second as interesting methods that may be employed in the HPC tools.

This book contains the contributed papers to the presentations at the workshop in September 2015.<sup>1</sup> As in the previous years, the workshop was organized jointly between the Center of Information Services and High Performance Computing (ZIH)<sup>2</sup> and the High Performance Computing Center (HLRS).<sup>3</sup>

Dresden, Germany  
January 2016

Andreas Knüpfer  
Tobias Hilbrich  
Christoph Niethammer  
José Gracia  
Wolfgang E. Nagel  
Michael M. Resch

---

<sup>1</sup><http://tools.zih.tu-dresden.de/2015/>.

<sup>2</sup><http://tu-dresden.de/zih/>.

<sup>3</sup><http://www.hlrs.de>.

# Contents

<b>1</b>	<b>Dyninst and MRNet: Foundational Infrastructure for Parallel Tools</b> . . . . .	<b>1</b>
	William R. Williams, Xiaozhu Meng, Benjamin Welton and Barton P. Miller	
<b>2</b>	<b>Validation of Hardware Events for Successful Performance Pattern Identification in High Performance Computing</b> . . . . .	<b>17</b>
	Thomas Röhl, Jan Eitzinger, Georg Hager and Gerhard Wellein	
<b>3</b>	<b>Performance Optimization for the Trinity RNA-Seq Assembler</b> . . . . .	<b>29</b>
	Michael Wagner, Ben Fulton and Robert Henschel	
<b>4</b>	<b>Power Management and Event Verification in PAPI</b> . . . . .	<b>41</b>
	Heike Jagode, Asim YarKhan, Anthony Danalis and Jack Dongarra	
<b>5</b>	<b>Gleaming the Cube: Online Performance Analysis and Visualization Using MALP</b> . . . . .	<b>53</b>
	Jean-Baptiste Besnard, Allen D. Malony, Sameer Shende, Marc Pérache and Julien Jaeger	
<b>6</b>	<b>Evaluation of Tool Interface Standards for Performance Analysis of OpenACC and OpenMP Programs</b> . . . . .	<b>67</b>
	Robert Dietrich, Ronny Tschüter, Tim Cramer, Guido Juckeland and Andreas Knüpfer	
<b>7</b>	<b>Extending MUST to Check Hybrid-Parallel Programs for Correctness Using the OpenMP Tools Interface</b> . . . . .	<b>85</b>
	Tim Cramer, Felix Münchhaffen, Christian Terboven, Tobias Hilbrich and Matthias S. Müller	
<b>8</b>	<b>Event Flow Graphs for MPI Performance Monitoring and Analysis</b> . . . . .	<b>103</b>
	Xavier Aguilar, Karl Furlinger and Erwin Laure	

- 9 Aura: A Flexible Dataflow Engine for Scalable Data Processing . . . . . 117**  
Tobias Herb, Lauritz Thamsen, Thomas Renner and Odej Kao
- 10 Parallel Code Analysis in HPC User Support . . . . . 127**  
Rene Sitt, Alexandra Feith and Dörte C. Sternel
- 11 PARCOACH Extension for Hybrid Applications with Interprocedural Analysis . . . . . 135**  
Emmanuelle Saillard, Hugo Brunie, Patrick Carribault and Denis Barthou
- 12 Enabling Model-Centric Debugging for Task-Based Programming Models—A Tasking Control Interface . . . . . 147**  
Mathias Nachtmann and José Gracia
- 13 Evaluating Out-of-Order Engine Limitations Using Uop Flow Simulation . . . . . 161**  
Vincent Palomares, David C. Wong, David J. Kuck and William Jalby