

# Contents

## I Integrated Development Environments

### **Sun HPC ClusterTools™ 7+: A Binary Distribution of Open MPI . . . . . 3**

Terry Dontje, Don Kerr, Daniel Lacher, Pak Lui, Ethan Mallove, Karen Norteman, Rolf Vandevaart, and Leonard Wisniewski

1	Introduction . . . . .	3
2	History . . . . .	4
3	Sun-Driven features . . . . .	5
4	Sun Product Activity . . . . .	13
5	Pros and Cons . . . . .	15
6	Future work and conclusions . . . . .	16
	References . . . . .	17

### **An Integrated Environment For the Development of Parallel Applications . . . . . 19**

Gregory R. Watson and Craig E. Rasmussen

1	Introduction . . . . .	19
2	Challenges . . . . .	21
3	Architecture . . . . .	23
4	A Simple Case Study . . . . .	28
5	Future Directions . . . . .	31
6	Conclusion . . . . .	33
	References . . . . .	34

### **Debugging MPI Programs on the Grid using g-Eclipse . . . . . 35**

Christof Klausecker, Thomas Köckerbauer, Robert Preissl, and Dieter Kranzlmüller

1	Introduction . . . . .	35
2	Related Work . . . . .	36
3	Overview of g-Eclipse Approach . . . . .	37
4	Remote Builder . . . . .	38

5	Grid Application Launchers .....	39
6	Trace Viewer .....	39
7	Conclusions and Future Work .....	44
	References .....	44

## II Parallel Communication and Debugging

<b>Enhanced Memory debugging of MPI-parallel Applications in Open MPI</b>		49
Shiqing Fan, Rainer Keller, and Michael Resch		
1	Introduction	49
2	Overview of Memcheck	50
3	Design and Implementation	51
4	Performance Implications	53
5	Detectable error classes and findings in actual applications	57
6	Conclusion and future work	59
	References	60
<b>MPI Correctness Checking with Marmot</b>		61
Bettina Krammer, Tobias Hilbrich, Valentin Himmler, Blasius Czink, Kiril Dichev, and Matthias S. Müller		
1	Introduction	62
2	Related Work	62
3	Design of Marmot	63
4	Collaboration with other tools	70
5	Experiences with real Applications	72
6	How to install and use Marmot	75
7	Conclusion and Future Work	76
	References	76
<b>Memory Debugging in Parallel and Distributed Applications</b>		79
Chris Gottbrath		
1	Introduction	79
2	The Challenges of Memory Debugging in Parallel Development	80
3	Classifying Memory Errors	80
4	Detecting Memory Leaks	82
5	The MemoryScape Debugger	82
6	MemoryScape Architecture	83
7	MemoryScape Features	84
8	MemoryScape Usage Tips	87
9	MemoryScape User Case Study: SIMULIA Uses MemoryScape to Find and Fix Bugs Quickly	88
10	Future MemoryScape Product Plans	90
11	Conclusion	90

### III Performance Analysis Tools

#### Sequential Performance Analysis with Callgrind and KCachegrind . . . . . 93

Josef Weidendorfer

1	Introduction . . . . .	93
2	Callgrind: a Call-Graph building Online Cache Simulator . . . . .	97
3	KCachegrind: Profile Visualization . . . . .	105
4	Usage Example . . . . .	110
5	Future Development . . . . .	111
	References . . . . .	113

#### Improving Cache Utilization Using Acumem VPE . . . . . 115

Erik Hagersten, Mats Nilsson and Magnus Vesterlund

1	Introduction . . . . .	116
2	Throughput Study of SPEC CPU 2006 . . . . .	118
3	First Generation Performance Tools Based on Hardware Counters . . . . .	120
4	Enter: The New Performance Tool . . . . .	122
5	Utilization Study of the Worst SPEC CPU 2006 Applications . . . .	126
6	Tuning Example: 179.art . . . . .	128
7	Tuning Example: Revisiting the Throughput Applications . . . . .	132
8	Conclusion . . . . .	134
	References . . . . .	135

### Parallel Performance Analysis Tools

#### The Vampir Performance Analysis Tool-Set . . . . . 139

Andreas Knüpfer, Holger Brunst, Jens Doleschal, Matthias Jurenz, Matthias

Lieber, Holger Mickler, Matthias S. Müller, and Wolfgang E. Nagel

1	Introduction . . . . .	139
2	Performance Analysis via Profiling or Tracing . . . . .	140
3	Instrumentation with VampirTrace . . . . .	141
4	Run-Time Measurement and Event Recording . . . . .	144
5	Trace Visualization with Vampir and VampirServer . . . . .	148
6	Related Work . . . . .	154
7	Conclusions and Future Work . . . . .	154
	References . . . . .	155

#### Usage of the SCALASCA toolset for scalable performance analysis of large-scale parallel applications . . . . . 157

Felix Wolf, Brian J. N. Wylie, Erika Ábrahám, Daniel Becker, Wolfgang

Frings, Karl Furlinger, Markus Geimer, Marc-André Hermanns, Bernd

Mohr, Shirley Moore, Matthias Pfeifer, and Zoltán Szebenyi

1	Introduction . . . . .	157
2	Overview . . . . .	158
3	Instrumentation and Measurement . . . . .	159

4	Trace Analysis .....	162
5	Understanding Performance Behavior .....	164
6	Outlook .....	166
	References .....	167
<b>Evolution of a Parallel Performance System .....</b>		<b>169</b>
Allen D. Malony, Sameer Shende, Alan Morris, Scott Biersdorff, Wyatt Spear, Kevin Huck, and Aroon Nataraj		
1	Introduction .....	169
2	TAU Performance System Design and Architecture .....	170
3	TAU Instrumentation .....	172
4	TAU Measurement .....	178
5	TAU Analysis .....	183
6	Conclusion and Future Work .....	186
	References .....	188
<b>Cray Performance Analysis Tools .....</b>		<b>191</b>
Luiz DeRose, Bill Homer, Dean Johnson, Steve Kaufmann, and Heidi Poxon		
1	Introduction .....	191
2	The Cray Performance Analysis Tools .....	192
3	Conclusions and Future Work .....	198
	References .....	199
<b>Index .....</b>		<b>201</b>

Tools for High Performance Computing

Proceedings of the 2nd International Workshop on  
Parallel Tools for High Performance Computing, July  
2008, HLRS, Stuttgart

Keller, R.; Himmler, V.; Krammer, B.; Schulz, A. (Eds.)

2008, XI, 202 p., Hardcover

ISBN: 978-3-540-68561-6