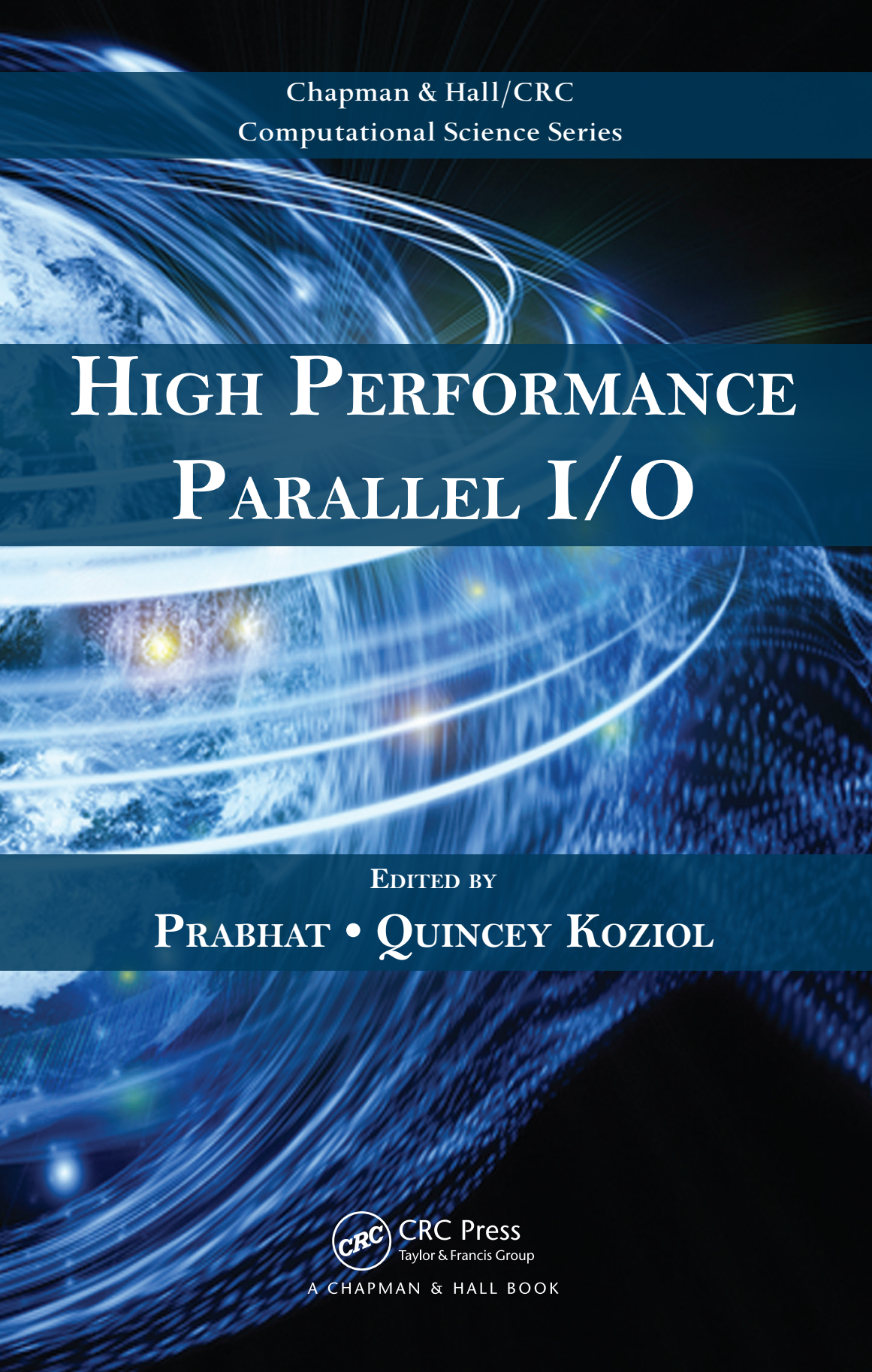




Chapman & Hall/CRC
Computational Science Series

HIGH PERFORMANCE PARALLEL I/O

EDITED BY

PRABHAT • QUINCEY KOZIOL



CRC Press
Taylor & Francis Group

A CHAPMAN & HALL BOOK

HIGH PERFORMANCE PARALLEL I/O

Chapman & Hall/CRC

Computational Science Series

SERIES EDITOR

Horst Simon

Deputy Director

Lawrence Berkeley National Laboratory

Berkeley, California, U.S.A.

PUBLISHED TITLES

COMBINATORIAL SCIENTIFIC COMPUTING

Edited by Uwe Naumann and Olaf Schenk

CONTEMPORARY HIGH PERFORMANCE COMPUTING: FROM PETASCALE
TOWARD EXASCALE

Edited by Jeffrey S. Vetter

DATA-INTENSIVE SCIENCE

Edited by Terence Critchlow and Kerstin Kleese van Dam

PETASCALE COMPUTING: ALGORITHMS AND APPLICATIONS

Edited by David A. Bader

FUNDAMENTALS OF MULTICORE SOFTWARE DEVELOPMENT

Edited by Victor Pankratius, Ali-Reza Adl-Tabatabai, and Walter Tichy

THE GREEN COMPUTING BOOK: TACKLING ENERGY EFFICIENCY AT LARGE SCALE

Edited by Wu-chun Feng

GRID COMPUTING: TECHNIQUES AND APPLICATIONS

Barry Wilkinson

HIGH PERFORMANCE COMPUTING: PROGRAMMING AND APPLICATIONS

John Levesque with Gene Wagenbreth

HIGH PERFORMANCE PARALLEL I/O

Prabhat and Quincey Koziol

HIGH PERFORMANCE VISUALIZATION:

ENABLING EXTREME-SCALE SCIENTIFIC INSIGHT

Edited by E. Wes Bethel, Hank Childs, and Charles Hansen

INTRODUCTION TO COMPUTATIONAL MODELING USING C AND
OPEN-SOURCE TOOLS

José M Garrido

INTRODUCTION TO CONCURRENCY IN PROGRAMMING LANGUAGES

Matthew J. Sottile, Timothy G. Mattson, and Craig E Rasmussen

PUBLISHED TITLES CONTINUED

INTRODUCTION TO ELEMENTARY COMPUTATIONAL MODELING: ESSENTIAL
CONCEPTS, PRINCIPLES, AND PROBLEM SOLVING

José M. Garrido

INTRODUCTION TO HIGH PERFORMANCE COMPUTING FOR SCIENTISTS
AND ENGINEERS

Georg Hager and Gerhard Wellein

INTRODUCTION TO REVERSIBLE COMPUTING

Kalyan S. Perumalla

INTRODUCTION TO SCHEDULING

Yves Robert and Frédéric Vivien

INTRODUCTION TO THE SIMULATION OF DYNAMICS USING SIMULINK®

Michael A. Gray

PEER-TO-PEER COMPUTING: APPLICATIONS, ARCHITECTURE, PROTOCOLS,
AND CHALLENGES

Yu-Kwong Ricky Kwok

PERFORMANCE TUNING OF SCIENTIFIC APPLICATIONS

Edited by David Bailey, Robert Lucas, and Samuel Williams

PROCESS ALGEBRA FOR PARALLEL AND DISTRIBUTED PROCESSING

Edited by Michael Alexander and William Gardner

SCIENTIFIC DATA MANAGEMENT: CHALLENGES, TECHNOLOGY, AND DEPLOYMENT

Edited by Arie Shoshani and Doron Rotem

HIGH PERFORMANCE PARALLEL I/O

EDITED BY

PRABHAT

Lawrence Berkeley National Laboratory
California, USA

QUINCEY KOZIOL

The HDF Group, Urbana-Champaign
Illinois, USA



CRC Press

Taylor & Francis Group
Boca Raton London New York

CRC Press is an imprint of the
Taylor & Francis Group, an **informa** business
A CHAPMAN & HALL BOOK

CRC Press
Taylor & Francis Group
6000 Broken Sound Parkway NW, Suite 300
Boca Raton, FL 33487-2742

© 2015 by Taylor & Francis Group, LLC
CRC Press is an imprint of Taylor & Francis Group, an Informa business

No claim to original U.S. Government works
Version Date: 20140915

International Standard Book Number-13: 978-1-4665-8235-4 (eBook - PDF)

This book contains information obtained from authentic and highly regarded sources. Reasonable efforts have been made to publish reliable data and information, but the author and publisher cannot assume responsibility for the validity of all materials or the consequences of their use. The authors and publishers have attempted to trace the copyright holders of all material reproduced in this publication and apologize to copyright holders if permission to publish in this form has not been obtained. If any copyright material has not been acknowledged please write and let us know so we may rectify in any future reprint.

Except as permitted under U.S. Copyright Law, no part of this book may be reprinted, reproduced, transmitted, or utilized in any form by any electronic, mechanical, or other means, now known or hereafter invented, including photocopying, microfilming, and recording, or in any information storage or retrieval system, without written permission from the publishers.

For permission to photocopy or use material electronically from this work, please access www.copyright.com (<http://www.copyright.com/>) or contact the Copyright Clearance Center, Inc. (CCC), 222 Rosewood Drive, Danvers, MA 01923, 978-750-8400. CCC is a not-for-profit organization that provides licenses and registration for a variety of users. For organizations that have been granted a photocopy license by the CCC, a separate system of payment has been arranged.

Trademark Notice: Product or corporate names may be trademarks or registered trademarks, and are used only for identification and explanation without intent to infringe.

Visit the Taylor & Francis Web site at
<http://www.taylorandfrancis.com>

and the CRC Press Web site at
<http://www.crcpress.com>

Contents

List of Figures	xix
List of Tables	xxv
Foreword	xxvii
Preface	xxix
Acknowledgments	xxxi
Contributors	xxxix
I Parallel I/O in Practice	1
1 Parallel I/O at HPC Facilities	3
<i>Galen Shipman</i>	
2 National Energy Research Scientific Computing Center	5
<i>Jason Hick</i>	
2.1 HPC at NERSC	5
2.2 I/O Hardware	6
2.2.1 Local Scratch File Systems	7
2.2.2 Storage Network	9
2.2.3 The NERSC Global File Systems	10
2.2.4 Archival Storage	11
2.3 Workflows, Workloads, and Applications	12
2.4 Conclusion	14
3 National Center for Supercomputing Applications	17
<i>William Kramer, Michelle Butler, Gregory Bauer, Kalyana Chadalavada, and Celso Mendes</i>	
3.1 The Blue Waters Computational and Analysis Subsystems .	18
3.2 Blue Waters On-line Storage Subsystem	19
3.2.1 On-line Storage Performance	22
3.3 Blue Waters Near-line Storage Subsystem and External Server Subsystem	24
3.4 Blue Waters Applications	28

3.4.1	Science and Engineering Team Application I/O Requirements	29
3.5	Conclusion	31
4	Argonne Leadership Computing Facility	33
	<i>William E. Allcock and Kevin Harms</i>	
4.1	HPC at ALCF	34
4.1.1	Intrepid	34
4.1.2	Mira	35
4.2	Overview of I/O at ALCF	35
4.3	I/O Hardware	36
4.3.1	Intrepid: ALCF Blue Gene/P System	37
4.3.2	Mira: ALCF Blue Gene/Q System	39
4.4	I/O Software	41
4.4.1	GPFS	41
4.4.1.1	Configuration	41
4.4.1.2	Tuning	42
4.4.1.3	Reliability	42
4.4.2	PVFS	43
4.4.3	Libraries	43
4.5	Workloads/Applications	44
4.5.1	Case Studies	46
4.6	Future I/O Plans at ALCF	47
5	Livermore Computing Center	51
	<i>Richard Hedges and Blaise Barney</i>	
5.1	Introduction	51
5.2	The Lustre® Parallel File System: Early Developments	53
5.3	Sequoia, Lustre® 2.0, and ZFS	54
5.4	IBM Blue Gene Systems	55
5.5	Sequoia File System Hardware	57
5.6	Experience with ZFS-Based Lustre® and Sequoia in Production	59
5.7	Sequoia I/O in Practice	60
5.7.1	General Remarks	60
5.7.2	Recommendations to Application Developers	61
5.7.3	SILO: LLNL's I/O Library	62
5.7.4	Scalable Checkpoint/Restart	62
5.8	Conclusion	63
6	Los Alamos National Laboratory	65
	<i>Gary Grider</i>	
6.1	HPC at LANL	65
6.1.1	Facilities and Environments	66
6.2	I/O Hardware	66

6.2.1	Storage Environment	67
6.2.2	Storage Area Networks	67
6.2.3	Global Parallel Scratch File Systems	68
6.2.4	The Curse of the Burst: Economic Thinking behind Burst Buffers	70
6.3	Workloads and Applications	72
6.3.1	Applications and Their Use of Storage	72
6.3.2	I/O Patterns and the Quest for Performance without Giving Up	73
6.3.3	Defeating N-to-1 Strided	73
6.4	Conclusion	75
7	Texas Advanced Computing Center	79
	<i>Karl W. Schulz</i>	
7.1	HPC at TACC	79
7.2	I/O Hardware	80
7.2.1	Performance	82
7.2.2	Parallel File Systems—A Shared Resource	84
7.3	Conclusion	86
II	File Systems	89
8	Lustre®	91
	<i>Eric Barton and Andreas Dilger</i>	
8.1	Motivation	91
8.2	Design and Architecture	92
8.2.1	Overview	92
8.2.2	Networking	93
8.2.2.1	LNet	93
8.2.2.2	RPC	94
8.2.3	Distributed Lock Manager	96
8.2.4	Back-end Storage	97
8.2.5	Metadata Server	98
8.2.6	Object Storage Server	99
8.2.7	Management Server	100
8.2.8	Client	100
8.2.9	Recovery	102
8.3	Deployment and Usage	103
8.4	Conclusion	104
9	GPFS	107
	<i>Dean Hildebrand and Frank Schmuck</i>	
9.1	Motivation	107
9.2	Design and Architecture	108
9.2.1	Shared Storage Model	108

9.2.2	Design Overview	110
9.2.3	Distributed Locking and Metadata Management . . .	111
9.2.3.1	The Distributed Lock Manager	111
9.2.3.2	Metadata Management	112
9.2.3.3	Concurrent Directory Updates	113
9.2.4	Advanced Data Management	114
9.2.4.1	GPFS Native RAID	114
9.2.4.2	Information Lifecycle Management	114
9.2.4.3	Wide-Area Caching and Replication	115
9.3	Deployment and Usage	116
9.3.1	Usage Examples	117
9.4	Conclusion	117
10	OrangeFS	119
	<i>Walt Ligon and Boyd Wilson</i>	
10.1	Motivation	120
10.1.1	PVFS1	120
10.1.2	PVFS2	121
10.1.3	OrangeFS	121
10.2	Design and Architecture	121
10.2.1	Overview	121
10.2.2	OrangeFS Request Protocol	122
10.2.3	File Structure Representation	122
10.2.3.1	Trove	123
10.2.4	Bulk Messaging Interface	123
10.2.5	Flows	124
10.2.6	Job Layer	124
10.2.7	Request State Machines	124
10.2.8	Distributed File Metadata	125
10.2.9	Distributed Directory Entry Metadata	125
10.2.10	Capability-Based Security	126
10.2.11	Clients and Interfaces	126
10.2.12	Features under Development	130
10.3	Deployment	131
10.3.1	Cluster Shared Scratch	132
10.3.2	Cluster Node Scratch	132
10.3.3	Amazon Web Services	133
10.4	Conclusion	133
11	OneFS	135
	<i>Nick Kirsch</i>	
11.1	Motivation	135
11.2	Design/Architecture	136
11.2.1	Isilon Node	137
11.3	Network	138

11.3.1	Back-End Network	138
11.3.2	Front-End Network	138
11.3.3	Complete Cluster View	139
11.4	OneFS Software Overview	139
11.4.1	Operating System	139
11.4.2	File System Structure	139
11.4.3	Data Layout	141
11.5	Data Protection	142
11.5.1	Power Loss	142
11.5.2	Scalable Rebuild	142
11.5.3	Virtual Hot Spare	143
11.5.4	$N + M$ Data Protection	143
11.6	Dynamic Scale/Scale on Demand	145
11.6.1	Performance and Capacity	145
11.7	Conclusion	147
III	I/O Libraries	149
12	I/O Libraries: Past, Present and Future	151
	<i>Mike Folk</i>	
12.1	Motivation	151
12.2	A Recent History of I/O Libraries, by Example	152
12.3	What Is the Future of I/O Libraries?	153
13	MPI-IO	155
	<i>Wei-keng Liao and Rajeev Thakur</i>	
13.1	Introduction	155
13.1.1	MPI-IO Background	156
13.1.2	Parallel I/O in Practice	156
13.2	Using MPI for Simple I/O	157
13.2.1	Three Ways of File Access	158
13.2.2	Blocking and Nonblocking I/O	159
13.3	File Access with User Intent	159
13.3.1	Independent I/O	160
13.3.2	MPI File View	161
13.3.3	Collective I/O	163
13.4	MPI-IO Hints	165
13.5	Conclusions	165
14	PLFS: Software-Defined Storage for HPC	169
	<i>John Bent</i>	
14.1	Motivation	169
14.2	Design/Architecture	170
14.2.1	PLFS Shared File Mode	170
14.2.2	PLFS Flat File Mode	172

14.2.3	PLFS Small File Mode	172
14.3	Deployment, Usage, and Applications	173
14.3.1	Burst Buffers	174
14.3.2	Cloud File Systems for HPC	174
14.4	Conclusion	175
15	Parallel-NetCDF	177
	<i>Rob Latham</i>	
15.1	Motivation	177
15.2	History and Background	179
15.3	Design and Architecture	179
15.4	Deployment and Usage	180
15.5	Additional Features	181
15.6	Conclusion	182
15.7	Additional Resources	183
16	HDF5	185
	<i>Quincey Koziol, Russ Rew, Mark Howison, Prabhat, and Marc Poinot</i>	
16.1	Motivation	186
16.2	History and Background	186
16.3	Design and Architecture	187
16.3.1	The HDF5 Data Model	188
16.3.2	The HDF5 Library	191
16.3.3	The HDF5 File Format	192
16.4	Usage and Applications	192
16.4.1	netCDF-4	193
16.4.1.1	Design	193
16.4.1.2	Applications	193
16.4.2	H5hut	194
16.4.2.1	Design	194
16.4.2.2	Applications	194
16.4.3	CGNS	195
16.4.3.1	Design	195
16.4.3.2	Applications	197
16.5	Conclusion	199
16.6	Additional Resources	199
17	ADIOS	203
	<i>Norbert Podhorszki, Scott Klasky, Qing Liu, Yuan Tian, Manish Parashar, Karsten Schwan, Matthew Wolf, and Sriram Lakshminarasimhan</i>	
17.1	Motivation	203
17.2	Design and Architecture	204
17.3	Deployment, Usage, and Applications	205
17.3.1	Checkpoint/Restart	205

17.3.2	Analysis	206
17.3.3	Code Coupling	207
17.3.4	Visualization	208
17.3.5	Data Reduction	210
17.3.6	Deployment	210
17.4	Conclusion	211
18	GLEAN	215
	<i>Venkatram Vishwanath, Huy Bui, Mark Hereld, and Michael E. Papka</i>	
18.1	Motivation	215
18.2	Design and Architecture	216
18.2.1	Exploiting Network Topology and Reduced Synchronization for I/O	217
18.2.2	Leveraging Application Data Semantics	218
18.2.3	Asynchronous Data Staging	219
18.2.4	Compression and Subfiling	219
18.3	Deployment, Usage, and Applications	220
18.3.1	Checkpoint, Restart, and Analysis I/O for HACC Cosmology	220
18.3.2	Data Staging for FLASH Astrophysics	221
18.3.3	Co-Visualization for PHASTA CFD Simulation	222
18.4	Conclusion	223
IV	I/O Case Studies	225
19	Parallel I/O for a Trillion-Particle Plasma Physics Simulation	227
	<i>Surendra Byna, Prabhat, Homa Karimabadi, and William Daughton</i>	
19.1	Abstract	227
19.2	Science Use Case	228
19.3	I/O Challenges	229
19.4	Software and Hardware	229
19.4.1	Hardware Platform	229
19.4.2	Software Setup	230
19.5	Parallel I/O in VPIC	230
19.6	Performance	232
19.6.1	Tuning Write Performance	232
19.6.2	HDF5 Tuning	233
19.6.3	Tuning Lustre File System and MPI-I/O Parameters	233
19.7	Conclusion	236
19.8	Acknowledgments	236

20 Stochastic Simulation Data Management	239
<i>Dimitris Servis</i>	
20.1 Background	239
20.2 Science Use Case	240
20.3 The I/O Challenge	241
20.4 Using HDF5 in Industrial Stochastic Simulations	243
20.4.1 Data Model and Versioning	244
20.4.2 VFL and Filters	244
20.4.3 Encryption	244
20.4.4 Robustness	244
20.4.5 Fragmentation	245
20.4.6 Process and Thread Synchronization	245
20.5 A (Near) Efficient Architecture Using HDF5	245
20.6 Performance	247
20.7 Conclusion	248
21 Silo: A General-Purpose API and Scientific Database	249
<i>Mark Miller</i>	
21.1 Canonical Use Case: ALE3D Restart and VisIt Visualization Workflow	250
21.2 Software, Hardware, and Performance	251
21.3 MIF and SSF Scalable I/O Paradigms	252
21.4 Successes with HDF5 as Middleware	256
21.5 Conclusion	257
22 Scaling Up Parallel I/O in S3D to 100-K Cores with ADIOS	259
<i>Scott Klasky, Gary Liu, Hasan Abbasi, Norbert Podhorszki, Jackie Chen, and Hemanth Kolla</i>	
22.1 Science Use Case	259
22.2 Software and Hardware	260
22.2.1 ADIOS-BP	261
22.2.2 Staged Write Method	262
22.2.3 Group-Based Hierarchical I/O Control	262
22.2.4 Aggregation and Subfiling	263
22.2.5 Index Generation	266
22.2.6 Staged Read Method	266
22.2.7 Staged Opens	266
22.2.8 Chunking	267
22.2.9 Limitations	268
22.3 Conclusion	269
23 In-Transit Processing: Data Analysis Using Burst Buffers	271
<i>Christopher Mitchell, David Bonnie, and Jonathan Woodring</i>	
23.1 Motivation	271
23.2 Design/Architecture	273

23.3 Systems Prototypes Related to Burst Buffers	274
23.4 Conclusion	275
V I/O Profiling Tools	277
24 Overview of I/O Benchmarking	279
<i>Katie Antypas and Yushu Yao</i>	
24.1 Introduction	279
24.2 I/O Benchmarking	280
24.3 Why Profile I/O in Scientific Applications?	283
24.4 Brief Introduction to I/O Profilers	283
24.5 I/O Profiling at NERSC	284
24.5.1 Application Profiling Case Studies	284
24.5.1.1 Checkpointing Too Frequently	285
24.5.1.2 Reading Small Input Files from Every Rank	286
24.5.1.3 Using the Wrong File System	286
24.6 Conclusion	287
25 TAU	289
<i>Sameer Shende and Allen D. Malony</i>	
25.1 Abstract	289
25.2 Features	290
25.2.1 MPI-IO Instrumentation	291
25.2.2 Runtime Preloading of Instrumented Library	291
25.2.3 Linker-Based Instrumentation	291
25.2.4 Instrumented External I/O Libraries	292
25.3 Success Stories	292
25.4 Conclusion	294
26 Integrated Performance Monitoring	297
<i>David Skinner</i>	
26.1 Design and Features	297
26.2 Success Stories	301
26.2.1 Chombo's ftruncate	301
26.2.2 MADBENCH and File System Health	302
26.2.3 Buffer Size	303
26.2.4 HPC Workload Studies	304
26.3 Conclusion	305
27 Darshan	309
<i>Philip Carns</i>	
27.1 Features	309
27.2 Success Stories	311
27.3 Conclusion	313

28 Iota	317
<i>Mark Howison, Prabhat, and Surendra Byna</i>	
28.1 Features	317
28.2 Success Stories	318
28.3 Conclusion	321
VI Future Trends	323
29 Parallel Computing Trends for the Coming Decade	325
<i>John Shalf</i>	
29.1 Technology Scaling	326
29.1.1 Classical Scaling Period (1965–2004)	326
29.1.2 End of Classical Scaling (2004)	326
29.1.3 Toward Data-Centric Computing (2014–2022)	328
29.2 Implications for the Future of Storage Systems	329
29.3 Conclusion	331
30 Storage Models: Past, Present, and Future	333
<i>Dries Kimpe and Robert Ross</i>	
30.1 The POSIX Era	334
30.2 The Current HPC Storage Model	335
30.2.1 The POSIX HPC I/O Extensions	335
30.2.2 MPI-IO	337
30.2.3 Object Storage Model	337
30.3 Post POSIX	338
30.3.1 Prior Work	338
30.3.2 Object Abstractions in HPC	339
30.3.3 Namespaces	341
30.4 Conclusion	341
31 Resilience	345
<i>Gary Grider and Nathan DeBardeleben</i>	
31.1 Present	346
31.1.1 Getting the Correct Answer	347
31.2 Future	348
31.3 Conclusion	350
32 Multi/Many Core	353
<i>Ramon Nou, Toni Cortes, Stelios Mavridis, Yannis Sfakianakis, and Angelos Bilas</i>	
32.1 Introduction	353
32.2 Storage I/O at Present	354
32.3 Storage I/O in the Near Future	355
32.4 Challenges and Solutions	356
32.4.1 NUMA Effects	356

32.4.2	Improving I/O Caching Efficiency	357
32.4.3	Dynamic I/O Scheduler Selection	359
32.5	Conclusion	360
33	Storage Networks and Interconnects	363
	<i>Parks Fields and Benjamin McClelland</i>	
33.1	Current State of Technology	363
33.2	Future Directions	365
33.3	Challenges and Solutions	366
33.4	Conclusion	367
34	Power Consumption	369
	<i>Matthew L. Curry, H. Lee Ward, David Martinez, Jill Gemmill, Jay Harris, Gary Grider, and Anna Maria Bailey</i>	
34.1	Introduction	370
34.2	Power Use in Recent and Current Supercomputers	370
34.2.1	Red Sky	371
34.2.2	Cielo	371
34.2.3	Palmetto	373
34.2.4	Dawn	374
34.2.5	Overall Survey Results	374
34.2.6	Extrapolation to Exascale	377
34.3	How I/O Changes at Exascale	377
34.3.1	Introducing More Asynchrony in the File System . . .	378
34.3.1.1	The Burst Buffer	378
34.3.1.2	Sirocco: A File System for Heterogeneous Media	378
34.3.2	Guarding against Single-Node Failures and Soft Errors	379
34.4	Conclusion	380
Index		385

List of Figures

2.1	NERSC compute and storage systems.	7
2.2	Edison local scratch diagram.	9
2.3	NERSC global file system architecture.	11
2.4	NERSC application codes.	13
3.1	The Blue Waters computational analysis subsystem.	20
3.2	The Blue Waters network subsystem.	21
3.3	The total Blue Waters I/O bandwidth is greater than 1.18 TB/s.	23
3.4	Blue Waters I/O performance tests scaled with the number of client nodes	24
3.5	The Blue Waters external server subsystem.	25
3.6	Snap-shot of jobs running on Blue Waters.	29
4.1	ALCF1 HPC center.	37
4.2	ALCF2 HPC system.	40
4.3	I/O interfaces by job size on Intrepid.	44
4.4	The amount of data transferred on <code>/intrepid-fs0</code> from 2010 to 2013.	45
4.5	File size CDF of project file systems.	46
4.6	Diagram of file system changes underway at ALCF.	48
5.1	Hardware packaging hierarchy of the BG/Q scaling architecture.	56
5.2	Hardware configuration of an OSS pair.	58
5.3	ZFS-based Lustre file system for LLN's Sequoia system.	59
6.1	LANL Parallel Scalable Backbone, as invented a decade ago.	69
6.2	Machine efficiency JMMTI over time for checkpoint.	70
6.3	Purchasing economics for disk, Flash, and hybrid storage for checkpoints.	71
6.4	PLFS internal data flow diagram.	74
6.5	PLFS N-to-1 application I/O speed-ups.	75
7.1	Lustre file system quotas and target usage on TACC's <i>Stampede</i> system.	81

7.2	Average software RAID performance for Lustre OST devices measured during disk burn-in procedure.	81
7.3	Lustre scaling across multiple Object Storage Servers (OSS) measured using Lustre (version 2.1.3).	82
7.4	Raw write performance characteristics of a large, parallel Lustre file system.	83
7.5	Daily file system performance measurements during normal production operations.	85
7.6	File system usage growth after starting production operations in January 2013.	86
8.1	Lustre architecture.	93
8.2	Lustre RPC.	95
9.1	GPFS configuration examples.	109
10.1	OrangeFS architecture.	122
10.2	OrangeFS: Certificate-based security.	127
10.3	OrangeFS: Clients and interfaces.	128
10.4	OrangeFS Direct interface.	129
10.5	Average job runtime vs. file system.	130
10.6	Average job runtime vs. number of remote clients.	131
11.1	OneFS distributed system.	137
11.2	All components of OneFS at work.	139
11.3	Single OneFS file system starting at the root inode <code>/ifs</code> . . .	140
11.4	OneFS: Data and parity in $N + M$ protect.	144
11.5	OneFS hybrid parity protection schemes ($N + M : x$). . . .	146
13.1	MPI-IO in the I/O software stack.	156
13.2	Simple example of a parallel file access pattern and sample MPI-IO code.	158
13.3	A 2D array partitioned among 4 MPI processes in a block-block pattern.	160
13.4	MPI program fragments to write the 2D array in parallel. .	161
13.5	Example of individual process's file views for 2D array. . . .	162
14.1	The architecture of PLFS shared file mode.	171
14.2	The architecture of PLFS small file mode.	173
14.3	PLFS enabled burst buffers.	175
15.1	The parallel I/O software stack.	178
15.2	The NetCDF file layout.	180
15.3	A code fragment demonstrating the two modes of the Parallel-NetCDF API.	181

15.4	A Parallel-NetCDF code fragment demonstrating the use of the non-blocking routines.	182
16.1	HDF5 data model.	189
16.2	HDF5 dataset.	189
16.3	HDF5 datatypes.	190
16.4	HDF5 program.	191
16.5	CGNS/SIDS data model.	196
16.6	CGNS file layout and parallel computation.	198
16.7	A graphical view of CGNS/HDF5 file.	198
17.1	ADIOS coupling workflow.	209
17.2	In-transit visualization using ADIOS.	209
18.1	Relationships between GLEAN and principal components of an HPC application.	217
18.2	Aggregation groups formed within a set of nodes by GLEAN.	218
18.3	Efficacy of topology-aware aggregation, subfiling, and compression on HACC I/O performance	221
19.1	VPIC simulation configured using MPI and OpenMP.	231
19.2	Parallel I/O stack and tunable parameters.	232
19.3	Performance improvement with patching HDF5 truncate.	234
19.4	VPIC-IOBench weak scaling study.	235
19.5	Performance of writing a single HDF5 file to Lustre.	235
20.1	An architecture for efficient stochastic simulation data management using HDF5.	246
21.1	Example of an ALE3D mesh decomposed into domains.	251
21.2	Example Silo-based MIF-IO.	253
21.3	Variation in ALE3D I/O performance as the number of file parts is varied.	254
21.4	Comparison of MIF-IO write and read performance on 3 ASC systems (Purple, Dawn and Graph) and improvement in I/O performance with block-based VFD.	257
22.1	Group-based hierarchical I/O control for ADIOS.	263
22.2	S3D write performance scaling with application size.	265
22.3	Metadata operation cost scaling with increasing number of writers.	267
22.4	S3D read performance scaling.	268
23.1	Potential system diagram for a supercomputer with burst buffers.	273

24.1	Application access patterns for moving data between memory and file.	281
24.2	Franklin XT-4 file system activity during an IOR run. . . .	282
24.3	Darshan statistics of a job that is checkpointing too frequently.	285
24.4	Hopper XE-6 file system activity during a user run. The user is frequently checkpointing, causing the I/O activity to be high.	285
24.5	Profile of a job spending its time in reading small input files.	286
24.6	Impact of using the wrong file system on job performance. .	286
25.1	I/O Profile for GCRM.	293
25.2	I/O profile for GCRM showing bytes for each file and read operation.	293
26.1	IPM's core framework.	298
26.2	IPM's double-open address hashing scheme.	299
26.3	Text output from an application run with IPM.	299
26.4	The POSIX and MPI-IO calls profiled by IPM's I/O module.	300
26.5	Steps required to improve performance of writes in HDF-based MPI codes.	302
26.6	Histograms of MADBENCH parallel reads before and after tuning the parallel file system.	303
26.7	Large numbers of jobs in a workload may be difficult to identify by name or other job metadata.	304
27.1	Excerpts from the darshan-job-summary utility included with Darshan.	311
27.2	Number of core hours consumed by Darshan-instrumented jobs in each partition size on Intrepid.	312
27.3	Total amount of data read and written by Darshan-instrumented jobs in each partition size category on Intrepid.	313
27.4	Prevalence of key I/O characteristics in each partition size category on Intrepid.	313
28.1	Overhead of the Iota library, tested with three I/O benchmarks.	319
28.2	Spatio-temporal pattern write patterns for the 12,000 core VPIC run.	320
28.3	Box and whisper plots show the distribution of write bandwidths to each OST in the 1,728 core VORPAL run.	320
29.1	The effect of the end of Dennard Scaling on microprocessor performance, power consumption, and architecture.	327
29.2	The Power and Clock Inflection point in 2004.	328
29.3	The energy consumed by data movement is starting to exceed the energy consumed by computation.	329

30.1	Domain decomposition resulting in highly non-contiguous access pattern.	336
30.2	Mapping of PnetCDF dataset to POSIX file, or EOF objects.	340
31.1	Estimates of the relative increase in error rates as a function of process technology.	349
31.2	Relationship between soft errors and voltage scaling through several process technology generations.	349
32.1	FIO read IOPS of native (XFS file system).	356
32.2	FIO read and write in a 64-core machine.	357
32.3	IOR read and write in a 64-core machine.	357
32.4	On-memory deduplication in file servers.	358
32.5	Dynamic partition on multicores.	359
32.6	IOAnalyzer system architecture.	360
33.1	IBTA InfiniBand roadmap.	364
33.2	Wavelength division multiplexing.	366
34.1	Predicted power use for Red Sky UC.	372
34.2	Power use for Palmetto and its storage infrastructure.	374
34.3	Absolute power use of Dawn over a one-month time period.	375
34.4	The percent of power used for disks and associated systems, by machine.	375
34.5	The number of disks installed in each system per teraFLOP of compute power, by machine.	376

List of Tables

2.1	NERSC storage systems noting storage type, bandwidth capabilities, and storage capacity.	8
2.2	Amount of archived data at NERSC in 2013 for major systems.	12
3.1	Blue Waters data payload bi-directional bandwidth in X, Y, Z dimensions.	20
3.2	Key Blue Waters performance information.	21
3.3	IOR-write performance on Blue Waters file systems.	23
3.4	IOR performance for writing 1-GB files on Blue Waters /scratch	24
3.5	PPM I/O performance.	31
4.1	Common I/O libraries at ALCF.	43
5.1	Performance characteristics of Blue Gene systems.	57
6.1	Detailed summary of LANL's 2013 computing environment.	67
6.2	Summary of the LANL storage environment.	67
23.1	Comparison of <i>In situ</i> , In Transit and conventional Post Processing methods.	272
34.1	Types of compute nodes used in Palmetto cluster.	373

Foreword

In the age of ever increasing emphasis on “big data” the topic of high performance parallel I/O should be amply covered in the literature. After all, what is the point of all the discussion of the ever increasing deluge of data without an understanding of how all the data are read and written quickly and efficiently? Yet, curiously a search on books on parallel I/O reveals that more than a decade has passed since any attempt has been made to summarize the state of our knowledge about high performance parallel I/O in the form of book for the High Performance Computing (HPC) community.

Prabhat and Quincey Koziol have made a significant accomplishment by editing this book on “High Performance Parallel I/O”. This book is indeed remarkable since there has been so much progress in technology, software, and tools for parallel I/O in the last decade, but documenting this progress has been notably absent. I/O is often the bottleneck to achieving the best possible performance in HPC, but its treatment and discussion are quite frequently secondary to the discussion of CPU performance. The authors have set out to address this deficiency, and succeeded admirably. Their text provides a useful overview of an area of rapid development that is currently not represented in any book.

An important distinction of the current book is that it has been edited by practitioners in the field. Both editors and most of the chapter authors have been involved hands-on in developing software for parallel I/O as well as porting and optimizing scientific applications on large scale parallel I/O system. This practical experience has led the editors to direct their selection of topics for the book towards a highly usable set of themes for the individual book sections: an overview of parallel I/O system as currently implemented in leading HPC centers, a survey of file systems and I/O libraries currently in use, and a selection of case studies, augmented by a description of tools for parallel I/O. The individual chapters are contributed by the leading experts in the field. Thus the book will be a handy reference for applications developer as well as for computer center managers, who want to know about the state-of-the-art in parallel I/O.

In the last twenty years high performance computing has seen many dramatic developments. In the early 1990s computing technology made a dramatic transition to MPPs using commodity hardware and the MPI programming model. This transition and the new parallel model for HPC solidified in the early 2000s. Essentially this model has remained the same until now, while

at the same time increasing performance by a factor of one million from the Gigaflops to the Petaflops level. Today we are close to yet another transformation of the HPC field as GPUs and accelerators become integrated, while the amount of parallelism seems to be ever increasing and the field is moving towards Exaflops level performance.

In this context of a potential rapid transformation of the high performance computing field, the book by Prabhat and Koziol arrives at exactly the right time. It succeeds perfectly and for the first time provides a survey of the significant accomplishments in file systems, libraries, and tools that have been developed for about a decade. These developments have now reached a state of relative maturity, and are ripe for a treatment in book format. Simultaneously the editors combine their technology and software survey with significant application development examples in a single volume. In the last set of chapters the book previews the I/O challenges in the Exaflops era. Thus the book will provide a solid foundation for anyone who is considering using the most recent tools for developing parallel I/O intensive applications today, and be also prepared for future Exascale platforms. I highly recommend this timely book for computational scientists and engineers.

Horst D. Simon
Lawrence Berkeley National Laboratory
Berkeley, September 2014

Preface

Parallel I/O is an integral component of modern high performance computing (HPC). Petascale-class simulations routinely produce terabyte to petabyte-sized datasets, which need to be stored efficiently. Data-centric analysis and visualization tools rely on efficient reads to ingest and process large datasets. Both write and read operations are critical for facilitating scientific discovery and insight.

This book captures the state of the art in the field of high performance parallel I/O in the 2013–2014 timeframe. We have drawn upon insights from leading practitioners, researchers, software architects, developers, and scientists. This rich tapestry of contributions from experts sheds light on the parallel I/O ecosystem.

The book is organized in six parts. Part I is intended to give readers a window into how large-scale HPC facilities scope, configure, and operate systems, with a specific emphasis on choices of I/O hardware, middleware, and applications. Readers will find leading storage experts from the National Energy Research Scientific Computing Center (NERSC), National Center for Supercomputing Applications (NCSA), Argonne Leadership Computing Facility (ALCF), Livermore Computing Center, and Texas Advanced Computing Center (TACC) sharing their perspectives in this part of the book.

Following this, the book traverses up the I/O software stack. In Part II, we deal with the file system layer. Leading designers and architects share their insights on the design, architecture, and application of the most prominent file systems in practice today: Lustre, GPFS, OrangeFS, and OneFS. Moving further up the I/O stack in part III, we review middleware (such as MPI-IO and PLFS), and user-facing libraries (such as Parallel-NetCDF, HDF5, ADIOS and GLEAN). These chapters give insight into design decisions made by library developers, library features, and applications.

Part IV of the book delves into real-world scientific applications that utilize the parallel I/O infrastructure. Application and library developers present case studies from particle-in-cell, stochastic, finite volume, and direct numerical simulations. Careful profiling and optimization are essential for obtaining peak (and sustained) parallel I/O performance. Part V of the book presents an overview of a number of profiling and benchmarking tools used by practitioners in the field. Finally, the world of HPC is forever in flux. Part VI of the book discusses implications of current trends in HPC on parallel I/O in the exascale world.

Acknowledgments

We would like to thank Mary Hester for an outstanding job in proofreading and editing the text. Her exemplary patience and diligence has vastly improved the quality of the book. Wes Bethel and Mike Folk encouraged us to take on this project. Randi Cohen from Taylor and Francis was very supportive of our endeavor and patient with our questions. Both of us have had a wonderful supportive environment in the Computational Research Division at Berkeley Lab, and the HDF Group, and we would like to acknowledge our colleagues. Finally, we would like to acknowledge the support and love offered by our families.

Prabhat, Quincey Koziol

April 15, 2014

Acknowledgments

The editors acknowledge the support rendered to them by the Lawrence Berkeley National Laboratory, and the HDF Group. This work was supported by the Director, Office of Science, Office and Advanced Scientific Computing Research, of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231.

Chapter 1—Parallel I/O at HPC Facilities

This work is sponsored by the U.S. Department of Energy, Office of Science, Advanced Scientific Computing Research program, and was performed at Oak Ridge National Laboratory (ORNL), which is managed by UT-Battelle, LLC, for the Department of Energy, under Contract No. DEAC0500OR22725.

Chapter 2—National Energy Research Scientific Computing Center

This work is supported by the Director, Office of Science, Office of Advanced Scientific Computing Research, of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231. This research used resources of the National Energy Research Scientific Computing Center (NERSC).

Chapter 3—National Center for Supercomputing Applications

This research is part of the Blue Waters sustained-petascale computing project, which is supported by the National Science Foundation (award number OCI 07-25070) and the state of Illinois. Blue Waters is a joint effort of the University of Illinois at Urbana-Champaign and its National Center for Supercomputing Applications.

Chapter 4—Argonne Leadership Computing Facility

This research used resources of the Argonne Leadership Computing Facility at Argonne National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under contract DE-AC02-06CH11357.

Chapter 5—Livermore Computing Center

The authors would like to acknowledge the entire Livermore Computing organization, which has supported the subject work for at least the last decade. Further, we acknowledge the efforts of the Lustre teams headed by Mark Gary, Chris Morrone, and Marc Stearman and the assistance of those three individuals in preparing and reviewing this manuscript.

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344. This chapter is an LLNL document: LLNL-BOOK-641234.

Chapter 6—Los Alamos National Laboratory

The author would like to acknowledge the entire Los Alamos High Performance Computing Division, which has supported the subject work for multiple decades. Los Alamos National Laboratory (LANL) is operated by Los Alamos National Security under its U. S. Department of Energy Contract No. DE-AC52-06NA25396.

Chapter 7—Texas Advanced Computing Center

This work was supported in part by the U.S. National Science Foundation under Award No. 1134872 from the Division of Cyberinfrastructure.

Chapter 8—Lustre

The authors' thanks go out to the US DOE for their initial funding and ongoing support for Lustre, along with Peter Braam, Mark Seager, and Gary Grider who started it all off.

Chapter 9—GPFS

GPFS is the result of the effort of a large number of very talented and dedicated people at IBM, all of whom have contributed to the design and implementation of GPFS over the last two decades. The authors would like to acknowledge the contributions of all of the members of the worldwide GPFS team to the work described here.

© Copyright IBM Corporation 2014. IBM, the IBM logo, and `ibm.com` are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. AIX, DB2, Power, and System x are registered trademarks of IBM Corporation. GPFS and Linear Tape File System are trademarks of IBM Corporation. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml (<http://www.ibm.com/>

`legal/copytrade.shtml`). Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both. Windows is a trademark of Microsoft Corporation in the United States, other countries, or both.

Chapter 11—OneFS

About Isilon: As the global leader in scale-out storage, Isilon delivers powerful yet simple solutions for enterprises that want to manage their data, not their storage. Isilon's products are simple to install, manage, and scale, at any size. And, unlike traditional enterprise storage, Isilon stays simple no matter how much storage is added, how much performance is required, or how business needs change in the future. Information about Isilon can be found at <http://www.isilon.com>. ©2011 Isilon Systems LLC. All rights reserved. Isilon, Isilon Systems, OneFS, and SyncIQ are registered trademarks of Isilon Systems LLC. Isilon IQ, SmartConnect, SnapshotIQ, TrueScale, Autobalance, FlexProtect, SmartCache, SmartPools, InsightIQ, "SIMPLE IS SMART," and the Isilon logo are trademarks of Isilon. Other product and company names mentioned are the trademarks of their respective owners. U.S. Patent Numbers 7,146,524; 7,346,720; 7,386,675. Other patents pending.

Chapter 13—MPI-IO

This work was supported in part by the U.S. Department of Energy under contracts DE-AC02-06CH11307 and DE-SC0005309 and in part by the U.S. National Science Foundation under Award No. CCF-0938000.

Chapter 14—PLFS: Software-Defined Storage for HPC

The author would like to acknowledge the many collaborators on PLFS: Garth Gibson, Milo Polte, and Chuck Cranor from CMU, Gary Grider, Meghan McClelland, Ben McClelland, Adam Manzanares, Aaron Torres, Brett Kettering, David Shrader, and Alfred Torrez from Los Alamos National Lab, and Sorin Faibish, Jingwang Zhang, Xuezhao Liu, Zhenhua Zhang, Percy Tzelnic, and Uday Gupta from EMC.

Chapter 15—Parallel-NetCDF

This research was supported by the Office of Science of the U.S. Department of Energy under Contract Nos. DE-AC02-05CH11231 and DE-AC02-06CH11357 including through the Scientific Discovery through Advanced Computing (SciDAC) Institute for Scalable Data Management, Analysis, and Visualization.

Chapter 16—HDF5

The authors would like to gratefully acknowledge the HDF5 development team at the HDF Group and others who have contributed to the HDF5 project.

Chapter 17—ADIOS

The authors would like to acknowledge the entire Oak Ridge Leadership Facility for their support for ADIOS. Oak Ridge National Laboratory (ORNL) is operated by UT-Battelle LLC under its U.S. Department of Energy Contract No. DE-AC05-00OR22725.

Chapter 18—GLEAN

We gratefully acknowledge the use of the resources of the Argonne Leadership Computing Facility at Argonne National Laboratory. This work was supported by the Office of Advanced Scientific Computing Research, Office of Science, U.S. Department of Energy, under Contract DE-AC02-06CH11357 including through the Scientific Discovery through Advanced Computing (SciDAC) Institute for Scalable Data Management, Analysis, and Visualization.

Chapter 19—Parallel I/O for a Trillion Particle Plasma Physics Simulation

This work is supported by the Director, Office of Science, Office of Advanced Scientific Computing Research, of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231. This research used resources of the National Energy Research Scientific Computing Center (NERSC).

Chapter 20—Stochastic Simulation Data Management

The author would like to thank Autoform for giving me the opportunity to experiment with HDF5 and The HDF Group for supporting my efforts.

Chapter 21—Silo

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory in part under Contract W-7405-Eng-48 and in part under Contract DE-AC52-07NA27344.

Chapter 22—Scaling up Parallel I/O in S3D to 100K Cores with ADIOS

The authors would like to acknowledge the entire Oak Ridge Leadership Facility for their support for ADIOS. Oak Ridge National Laboratory (ORNL)

is operated by UT-Battelle LLC under its U.S. Department of Energy Contract No. DE-AC05-00OR22725 including the Scientific Discovery through Advanced Computing (SciDAC) Institute for Scalable Data Management, Analysis, and Visualization.

Chapter 23—In-Transit Processing: Data Analysis Using Burst Buffers

The authors would like to thank the Los Alamos National Laboratory's Laboratory Directed Research and Development (LDRD) program for its support of this research under project 20130457ER: Co-Design of Burst Buffer Hardware and Data Analysis/Visualization Software for Large-Scale Simulations. Los Alamos National Laboratory (LANL) is operated by Los Alamos National Security under its U.S. Department of Energy Contract No. DE-AC52-06NA25396. This chapter was reviewed for release under Los Alamos National Laboratory LA-UR-13-27966.

Chapter 24—Overview of I/O Benchmarking

This work used resources of the National Energy Research Scientific Computing Center, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231.

Chapter 25—TAU

The research at the University of Oregon was supported by grants ER26057, ER26167, ER26098, and ER26005 from the U.S. Department of Energy, Office of Science.

Chapter 26—Integrated Performance Monitoring

Integrated Performance Monitoring (IPM) extends earlier work by David Skinner and the ACTC group at IBM's TJ Watson Research Center. IPM is supported by the National Science Foundation (NSF0721397) and the Department of Energy Office of Science (DE-AC02-05CH11231). The authors thank Bill Kramer for his input on IPM's design goals.

Chapter 27—Darshan

This research was supported by the Office of Science of the U.S. Department of Energy under Contract Nos. DE-AC02-05CH11231 and DE-AC02-06CH11357 including through the Scientific Discovery through Advanced Computing (SciDAC) Institute for Scalable Data Management, Analysis, and Visualization.

Chapter 28—Iota

Iota is an extension of earlier work by Noel Keen and Karl Fuerlinger to incorporate I/O tracing into the Integrated Performance Monitoring (IPM) tool.

This research was conducted using computational resources and services at the Center for Computation and Visualization, Brown University. This work was supported by the Director, Office of Science, Office of Advanced Scientific Computing Research, of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231. This research used resources of the National Energy Research Scientific Computing Center.

Chapter 29—Parallel Computing Trends for the Coming Decade

This work was supported by the ASCR Office in the DOE Office of Science under contract number DE-AC02-05CH11231. This work also has benefitted greatly from collaborator Peter Kogge whose long-term data collection on emerging trends in CMOS silicon technology have been the underpinning for the DARPA and DOE programs in exascale computing.

Chapter 30—Storage Models: Past, Present, and Future

The EOF work and the writing of this chapter were supported by the U.S. Department of Energy, under Contract DE-AC02-06CH11357.

Chapter 32—Multi/Many Core

We thankfully acknowledge the support of the European Commission under the 7th Framework Programs through the IOLANES (FP7-ICT-248615) and HiPEAC3 (FP7-ICT-287759) projects, the Spanish Ministry of Economy and Competitiveness under the TIN2012-34557 grant, and by the Catalan Government under the 2009-SGR-980 grant.

Chapter 33—Storage Networks and Interconnects

The authors would like to acknowledge the Department of Energy, the National Nuclear Security Administration, and our employer Los Alamos National Laboratory.

Chapter 34—Power Consumption

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed

Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Disclaimer

This document was prepared as an account of work sponsored by the United States government. While this document is believed to contain correct information, neither the United States government nor any agency thereof, nor the Regents of the University of California, nor any of their employees, makes any warranty, express or implied, or assumes any legal responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by its trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States government or any agency thereof, or the Regents of the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States government or any agency thereof or the Regents of the University of California.

Contributors

Hasan Abbasi

Oak Ridge National Laboratory
Oak Ridge, TN, USA

William E. Allcock

Argonne National Laboratory
Argonne, IL, USA

Katie Antypas

National Energy Research Scientific
Computing Center
Oakland, CA, USA

Anna Maria Bailey

Lawrence Livermore National
Laboratory
Livermore, CA, USA

Blaise Barney

Lawrence Livermore National
Laboratory
Livermore, CA, USA

Eric Barton

Intel Corporation
Bristol, UK

Gregory Bauer

National Center for Supercomputing
Applications
Urbana, IL, USA

John Bent

EMC Corporation
USA

Angelos Bilas

Foundation for Research and
Technology–Hellas
Heraklion, Greece

David Bonnie

Los Alamos National Laboratory
Los Alamos, NM, USA

Huy Bui

Argonne National Laboratory
Argonne, IL, USA

Michelle Butler

National Center for Supercomputing
Applications
Urbana, IL, USA

Surendra Byna

Lawrence Berkeley National
Laboratory
Berkeley, CA, USA

Philip Carns

Argonne National Laboratory
Argonne, IL, USA

Kalyana Chadalavada

National Center for Supercomputing
Applications
Urbana, IL, USA

Jackie Chen

Sandia National Laboratory
Livermore, CA, USA

Toni Cortes

Barcelona Supercomputing Center
Barcelona, Spain

Matthew L. Curry

Sandia National Laboratories
Albuquerque, NM, USA

William Daughton

Los Alamos National Laboratory
Los Alamos, NM, USA

Nathan DeBardeleben

Los Alamos National Laboratory
Los Alamos, NM, USA

Andreas Dilger

Intel Corporation
Calgary, Canada

Parks Fields

Los Alamos National Laboratory
Los Alamos, NM, USA

Mike Folk

The HDF Group
Champaign, IL, USA

Jill Gemmill

Clemson University
Clemson, SC, USA

Garth Gibson

Carnegie Mellon University
Pittsburgh, PA, USA

Gary Grider

Los Alamos National Laboratory
Los Alamos, NM, USA

Kevin Harms

Argonne National Laboratory
Argonne, IL, USA

Jay Harris

Clemson University
Clemson, SC, USA

Richard Hedges

Lawrence Livermore National
Laboratory
Livermore, CA, USA

Mark Hereld

Argonne National Laboratory
Argonne, IL, USA

Jason Hick

National Energy Research Scientific
Computing Center
Oakland, CA, USA

Dean Hildebrand

IBM Almaden Research Center
San Jose, CA, USA

Mark Howison

Brown University
Providence, RI, USA

Homa Karimabadi

University of California
San Diego, CA, USA

Dries Kimpe

Argonne National Laboratory
Argonne, IL, USA

Nick Kirsch

Isilon
Seattle, WA, USA

Scott Klasky

Oak Ridge National Laboratory
Oak Ridge, TN, USA

Hemanth Kolla

Sandia National Laboratory
Livermore, CA, USA

Quincey Koziol

The HDF Group
Champaign, IL, USA

Bill Kramer

National Center for Supercomputing
Applications
Urbana, IL, USA

Sriram Lakshminarasimhan

North Carolina State University
Raleigh, NC, USA

Rob Latham

Argonne National Laboratory
Argonne, IL, USA

Wei-keng Liao

Northwestern University
Evanston, IL, USA

Walt Ligon

Clemson University
Clemson, SC, USA

Qing Liu

Oak Ridge National Laboratory
Oak Ridge, TN, USA

Allen Maloney

University of Oregon
Eugene, OR, USA

David Martinez

Sandia National Laboratories
Albuquerque, NM, USA

Stelios Mavridis

Foundation for Research and
Technology–Hellas
Heraklion, Greece

Benjamin McClelland

Los Alamos National Laboratory
Los Alamos, NM, USA

Celso Mendes

National Center for Supercomputing
Applications
Urbana, IL, USA

Mark Miller

Lawrence Livermore National
Laboratory
Livermore, CA, USA

Christopher Mitchell

Los Alamos National Laboratory
Los Alamos, NM, USA

Ramon Nou

Barcelona Supercomputing Center
Barcelona, Spain

Michael E. Papka

Argonne National Laboratory
Argonne, IL, USA

Manish Parashar

Rutgers University
Piscataway, NJ, USA

Norbert Podhorszki

Oak Ridge National Laboratory
Oak Ridge, TN, USA

Marc Poinot

ONERA: French Aerospace Lab
Châtillon, France

Prabhat

Lawrence Berkeley National
Laboratory
Berkeley, CA, USA

Russ Rew

University Corporation for
Atmospheric Research
Boulder, CO, USA

Rob Ross

Argonne National Laboratory
Argonne, IL, USA

Frank Schmuck

IBM Almaden Research Center
San Jose, CA, USA

Karl W. Schulz

Texas Advanced Computing Center
Austin, TX, USA

Karsten Schwan

Georgia Institute of Technology
Atlanta, GA, USA

Dimitris Servis

AutoForm Development GmbH
Zurich, Switzerland

Yannis Sfakianakis

Foundation for Research and
Technology–Hellas
Heraklion, Greece

John Shalf

Lawrence Berkeley National
Laboratory
Berkeley, CA, USA

Sameer Shende

University of Oregon
Eugene, OR, USA

Galen Shipman

Oak Ridge National Laboratory
Oak Ridge, TN, USA

David Skinner

National Energy Research Scientific
Computing Center
Oakland, CA, USA

Rajeev Thakur

Argonne National Laboratory
Argonne, IL, USA

Yuan Tian

Oak Ridge National Laboratory
Oak Ridge, TN, USA

Venkat Vishwanath

Argonne National Laboratory
Argonne, IL, USA

H. Lee Ward

Sandia National Laboratories
Albuquerque, NM, USA

Boyd Wilson

Omnibond Systems LLC
Pendleton, SC, USA

Matthew Wolf

Georgia Institute of Technology
Atlanta, GA, USA

Jonathan Woodring

Los Alamos National Laboratory
Los Alamos, NM, USA

Yushu Yao

National Energy Research Scientific
Computing Center
Oakland, CA, USA

Part I

Parallel I/O in Practice