

---

**INPUT/OUTPUT IN  
PARALLEL AND DISTRIBUTED  
COMPUTER SYSTEMS**

---

**THE KLUWER INTERNATIONAL SERIES  
IN ENGINEERING AND COMPUTER SCIENCE**

---

# **INPUT/OUTPUT IN PARALLEL AND DISTRIBUTED COMPUTER SYSTEMS**

*edited by*

**Ravi Jain**

*Bell Communications Research  
Morristown, New Jersey, USA*

**John Werth**

*University of Texas at Austin  
Austin, Texas, USA*

**James C. Browne**

*University of Teas at Austin  
Austin, Texas, USA*



**KLUWER ACADEMIC PUBLISHERS**  
**Boston / Dordrecht / London**

---

**Distributors for North America:**

Kluwer Academic Publishers  
101 Philip Drive  
Assinippi Park  
Norwell, Massachusetts 02061 USA

**Distributors for all other countries:**

Kluwer Academic Publishers Group  
Distribution Centre  
Post Office Box 322  
3300 AH Dordrecht, THE NETHERLANDS

---

**Library of Congress Cataloging-in-Publication Data**

A C.I.P. Catalogue record for this book is available  
from the Library of Congress.

---

ISBN-13: 978-1-4612-8607-3      e-ISBN-13: 978-1-4613-1401-1  
DOI: 10.1007/978-1-4613-1401-1

**Copyright © 1996 by Kluwer Academic Publishers**

Softcover reprint of the hardcover 1st edition 1996

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, mechanical, photocopying, recording, or otherwise, without the prior written permission of the publisher, Kluwer Academic Publishers, 101 Philip Drive, Assinippi Park, Norwell, Massachusetts 02061

*Printed on acid-free paper.*

*To*

*Meera, Laurie and Gayle*

---

# CONTENTS

<b>PREFACE</b>	xiii
<b>Part I INTRODUCTION</b>	1
<b>1 I/O IN PARALLEL AND DISTRIBUTED SYSTEMS: AN INTRODUCTION</b>	
<i>Ravi Jain, John Werth and J. C. Browne</i>	3
1 Introduction	3
2 Survey of I/O issues	5
3 Trends and emerging concerns	15
4 Summary	20
<b>2 AN INTRODUCTION TO PARALLEL I/O MODELS AND ALGORITHMS</b>	
<i>Elizabeth Shriver and Mark Nodine</i>	31
1 Introduction	32
2 The Parallel Disk Model	34
3 Parallel Disk Model algorithms	42
4 Other two-level parallel-I/O memory models and their algorithms	56
5 Related systems approaches	58
6 Conclusions	60
<b>3 ISSUES IN COMPILING I/O INTENSIVE PROBLEMS</b>	
<i>Rajesh Bordawekar and Alok Choudhary</i>	69
1 Introduction	69
2 Architectural Model	70

3	Programming Model	73
4	Working spaces in I/O Intensive Parallel Programs	75
5	Execution Models	79
6	Compiling Out-of-core Parallel Programs	83
7	Summary	94
<b>4</b>	<b>INTRODUCTION TO MULTIPROCESSOR I/O ARCHITECTURE</b>	
	<i>David Kotz</i>	97
1	Introduction	97
2	Review and Terminology	98
3	Example architectures	100
4	Disk I/O	104
5	Tape I/O	116
6	Graphics I/O	117
7	Network I/O	117
8	Summary	118
<b>Part II</b>	<b>SYSTEM SOFTWARE</b>	125
<b>5</b>	<b>OVERVIEW OF THE MPI-IO PARALLEL I/O INTERFACE</b>	
	<i>Peter Corbett, Dror Feitelson, Sam Fineberg, Yarsun Hsu, Bill Nitzberg, Jean-Pierre Prost, Marc Snir, Bernard Traversat, and Parkson Wong</i>	127
1	Parallel I/O	128
2	Overview of MPI-IO	130
3	Data Partitioning in MPI-IO	131
4	MPI-IO Data Access Functions	134
5	Miscellaneous Features	140
6	Current Status	141
	APPENDIX A Transposing a 2-D Matrix	141
	REFERENCES	143

## Contents

### **6 RUNTIME SUPPORT FOR OUT-OF-CORE PARALLEL PROGRAMS**

<i>Rajeev Thakur and Alok Choudhary</i>	147
1 Introduction	147
2 Motivating Example	148
3 Extended Two-Phase Method	150
4 Partitioning I/O Among Processors	154
5 Performance	157
6 Advantages	161
7 Conclusions	163

### **7 PARALLEL I/O WORKLOAD CHARACTERISTICS USING VESTA**

<i>Sandra Johnson Baylor and C. Eric Wu</i>	167
1 Introduction	167
2 Architecture and Parallel File System	169
3 Applications	170
4 Methodology	173
5 Results	175
6 Conclusion	182

### **8 VIDEO ON DEMAND USING THE VESTA PARALLEL FILE SYSTEM**

<i>Edgar T. Kalns and Yarsun Hsu</i>	187
1 Introduction	187
2 Related Work	188
3 Vesta Parallel File System Overview	191
4 VoD Experimentation Environment	191
5 Vesta VoD Performance	195
6 Conclusion	202

### **9 LOW-LEVEL INTERFACES FOR HIGH-LEVEL PARALLEL I/O**

<i>Nils Nieuwejaar and David Kotz</i>	205
1 Introduction	205
2 The Conventional Interface	206



3	Access Patterns	207
4	File System Interfaces	211
5	Other Unconventional Interfaces	219
6	Conclusion	221
<b>10</b>	<b>SCALABLE CONCURRENCY CONTROL FOR PARALLEL FILE SYSTEMS</b>	
	<i>Steven A. Moyer and V. S. Sunderam</i>	225
1	Introduction	225
2	Volatile Transactions	227
3	Implementation	228
4	Observations	231
5	Experimental Results	232
6	Related Work	238
7	Conclusions	239
	APPENDIX A      Deadlock Avoidance with Progress	240
<b>11</b>	<b>IMPROVING THE PERFORMANCE OF PARALLEL I/O USING DISTRIBUTED SCHEDULING ALGORITHMS</b>	
	<i>Dannie Durand, Ravi Jain and David Tseytlin</i>	245
1	Introduction	246
2	Background	246
3	Problem Description	248
4	A Distributed Scheduling Algorithm	251
5	Experimental Results	256
6	Extensions	263
7	Conclusions	264
	APPENDIX A      A bound on the number of holes	268
<b>12</b>	<b>PLACEMENT-RELATED PROBLEMS IN SHARED DISK I/O</b>	
	<i>J.B. Sinclair, J. Tang and P.J. Varman</i>	271
1	Introduction	271
2	External Merging	273
3	Analysis of a 2-Disk System	278

## Contents

4	Analysis of a Multi-Disk System	282
5	Solutions to Racing	285
6	Summary	288
<b>Part III</b>	<b>ARCHITECTURE</b>	<b>291</b>
<b>13</b>	<b>PERFORMANCE EVALUATION OF A MASSIVELY PARALLEL I/O SUBSYSTEM</b>	
	<i>Sandra Johnson Baylor, Caroline Benveniste, and Yarsun Hsu</i>	293
1	Introduction	293
2	The Vulcan Architecture	295
3	Simulation Methodology	299
4	Results	302
5	Conclusion	310
<b>14</b>	<b>HETEROGENEOUS I/O CONTENTION IN A SINGLE-BUS MULTIPROCESSOR</b>	
	<i>Steven H. VanderLeest and Ravishankar K. Iyer</i>	313
1	Introduction	313
2	Related Work	314
3	Description of Experiment	315
4	Preliminary Analysis	320
5	The Performance Impact of I/O Contention	323
6	Concluding Remarks	329
<b>15</b>	<b>HCSA: A HYBRID CLIENT-SERVER ARCHITECTURE</b>	
	<i>Gerhard A. Schloss and Michael Vernick</i>	333
1	Introduction	333
2	Architectures	334
3	Hybrid Client-Server Architecture	336
4	File Access Protocols	339
5	HCSA Performance Study	341
6	Conclusions	348
	REFERENCES	349

<b>16</b>	<b>A SCALABLE DISK SYSTEM WITH DATA RECONSTRUCTION FUNCTIONS</b>	
	<i>Haruo Yokota and Yasuyuki Mimatsu</i>	353
1	Introduction	353
2	Applying Parity Technique on an Interconnection Network	356
3	Estimation of the Response Time and Throughput	360
4	An Experimental System and Performance Evaluation	364
5	Discussion on the Reliability	368
6	Concluding Remarks	370
<b>17</b>	<b>AN EXPERIMENTAL MEMORY-BASED I/O SUBSYSTEM</b>	
	<i>Abhaya Asthana and Mark Cravatts and Paul Krzyzanowski</i>	373
1	Introduction	373
2	SWIM active memory	374
3	System architecture	376
4	An object based programming model	377
5	Built-in support mechanisms	379
6	Prototype status	382
7	Application examples	383
8	Conclusion	389
	<b>INDEX</b>	391

---

# PREFACE

I/O for parallel and distributed computer systems has drawn increasing attention over the last few years as it has become apparent that I/O performance, rather than CPU performance, may be the key limiting factor in the performance of future systems. This *I/O bottleneck* is caused by the increasing speed mismatch between processing units and storage devices, the use of multiple processors operating simultaneously in parallel and distributed systems, and by the increasing I/O demands of new classes of applications, like multimedia. It is also important to note that, to varying degrees, the I/O bottleneck exists at multiple levels of the memory hierarchy. All indications are that the I/O bottleneck will be with us for some time to come, and is likely to increase in importance.

These realizations prompted us to advocate that the I/O bottleneck be addressed systematically at all levels of parallel and distributed system design. Thus while there are solutions which focus on one aspect of the system (e.g., architectural solutions like RAID), we felt that the benefits of these solutions would not be realized unless I/O-efficient design was integrated into applications, algorithms, compilers, operating systems, and architectures for parallel and distributed systems. With this view we initiated, in 1993, a workshop dedicated to I/O in parallel and distributed systems, held in conjunction with the International Parallel Processing Symposium (IPPS). The workshop drew substantial interest, and has now become an annual event. Papers submitted to the workshop were refereed, and in 1994 and 1995 about 30%-40% of the submitted papers were accepted for presentation. In 1996 the workshop on I/O in Parallel and Distributed Systems (IOPADS) is being held as an independent workshop with the Federated Computing Research Conference, and continues to draw international research participation.

This book is divided into three parts. Part I, the Introduction, contains four invited chapters which provide a tutorial and survey of I/O issues in parallel and distributed systems. The chapters in Parts II and III contain selected research papers from the 1994 and 1995 IOPADS workshops; many of these papers have been substantially revised and updated for inclusion in this volume. Part II collects the papers from both years which dealt with various aspects of system software, and Part III those addressing primarily architectural issues.

The first chapter in Part I provides an overview of I/O issues, and surveys upcoming trends in this area, such as the convergence of networking and I/O, the increasing importance of the Internet and World-Wide Web as a new level of the system memory hierarchy, and the challenges posed by mobile and wireless computing. Chapter 2 by Shriver and Nodine presents an introduction to parallel I/O models and algorithms. Chapter 3 by Thakur and Choudhary, and Chapter 4 by Kotz, provide introductions to compiler issues and architectural approaches, respectively. We hope Part I will prove a useful source for graduate students and researchers new to the area.

The papers in Part II include those on compiler support, programming models, interfaces, file systems and scheduling. Chapter 5, by a team from IBM and NASA, presents the MPI-IO parallel I/O interface, and Chapter 6 by Thakur and Choudhary describes work on runtime support for out-of-core data parallel algorithms. Chapter 7 by Baylor et al presents a summary of the Vesta parallel file system developed at IBM, and its use for collecting parallel I/O workload characteristics, while Chapter 8 by Kalns and Hsu describes its use for an interesting and important application, video-on-demand. In Chapter 9, Nieuwejaar and Kotz present the results of tracing a parallel file system for scientific applications; the results indicate extensions needed to the interface provided to the programmer. Moyer and Sunderam, in Chapter 10, address the concurrency control issues that can arise when even a single read or write operation results in parallel I/O operations on multiple storage devices. In Chapter 11, Durand et al present distributed algorithms for scheduling parallel I/O operations so as to minimize their completion time. In Chapter 12, Sinclair et al describe scenarios in which multiple processes engaging in parallel I/O can conflict such that a subset of them monopolize the I/O resources; this work has implications for data allocation and task partitioning in parallel systems.

Part III of the book focuses on issues relating to system architecture. Chapter 13, by Baylor et al of IBM, describes a performance evaluation of the massively parallel I/O subsystem of the Vulcan MPP, with results on the placement of I/O nodes in the system. VanderLeest and Iyer present, in Chapter 14, a methodology for measuring bus contention, a critical resource in parallel I/O, and the results of such a study on a specific system. In Chapter 15, Schloss and Vernick present a Hybrid Client-Server Architecture, in which the traditional client-server architecture is modified to give clients access to both the server and its disks. In Chapter 16, Yokota and Mimatsu propose Data-reconstruction networks for I/O subsystems, where each node has a set of disks and nodes are interconnected by a network separate from the primary processor interconnection network. Finally, in Chapter 17, Asthana et al describe an I/O subsystem in which processing logic is associated with each memory chip, offloading some low-level I/O-related tasks from the CPU and thus speeding up overall system operation.

## *Preface*

The depth and breadth of the chapters in Parts II and III indicates the vitality of this fast-growing research area, and we hope they will stimulate further study and integration of approaches for alleviating the I/O bottleneck.

We would like to thank the members of the IOPADS 1994 and 1995 program committees who reviewed papers. The program committee members were: Abhaya Asthana (AT&T Bell Labs), Larry Berdahl (Lawrence Livermore), Peter Chen (Univ. of Michigan), Alok Choudhary (Syracuse), Peter Corbett (IBM Watson), Tom Cormen (Dartmouth), David DeWitt (Univ. of Wisconsin), Sam Fineberg (NASA Ames), Shahram Ghandeharizadeh (USC), Paul Messina (Caltech), John Nickolls (MasPar), and Wayne Roiger (Cray Research). We also thank the many other reviewers who provided us with reviews of the submitted papers. Papers from the 1994 Workshop also appeared, in condensed form, in the ACM SIGARCH newsletter "Computer Architecture News", Oct. 1994; thanks are due to Doug DeGroot, editor, for his help in that regard. We thank Scott Delman of Kluwer for his help and patience during the long process of preparing this book, and Ravi Jain would like to thank Michael Kramer of Bellcore for his support.

Finally, we would like to thank the organizers of IPPS, and especially IPPS Chair Viktor Prasanna, for the opportunity to hold the IOPADS workshops in order to bring together researchers addressing the I/O issues in parallel and distributed systems.

*Ravi Jain, John Werth and J. C. Browne*