https://dl.acm.org/doi/10.1145/224538.224638 120%

ACM DIGITAL LIBRARY | Association for Computing Machinery

Browse | About | Sign in | Register

Journals | Magazines | Proceedings | Books | SIGs | Conferences | People

Search ACM Digital Library | Advanced Search

Conference | Proceedings | Upcoming Events | Authors | Affiliations | Award Winners

Feedback

Home > Conferences > ICS > Proceedings > ICS '95 > PPFS: a high performance portable parallel file system

ARTICLE | FREE ACCESS

PPFS: a high performance portable parallel file system

Authors: James V. Huber, Jr., Andrew A. Chien, Christopher L. Elford, David S. Blumenthal, Daniel A. Reed | Authors Info & Claims

ICS '95: Proceedings of the 9th international conference on Supercomputing • Pages 385 - 394 • https://doi.org/10.1145/224538.224638

Published: 03 July 1995 Publication History

Check for updates

132 830

PDF | eReader

ICS

PPFS: a high performance portable parallel file system

Authors: James V. Huber, Jr., Andrew A. Chien, Christopher L. Elford, David S. Blumenthal, Daniel A. Reed Authors Info & Claims

ICS '95: Proceedings of the 9th international conference on Supercomputing Pages 385 - 394
https://doi.org/10.1145/224538.224638

Published: 03 July 1995 Publication History

Check for updates

132 828

PDF eReader

Feedback

27

PDF

Help

James V. Huber, Jr.* Christopher L. Elford* Daniel A. Reed*
Andrew A. Chien* David S. Blumenthal*

Department of Computer Science

ICS

Abstract

Rapid increases in processor performance over the past decade have outstripped performance improvements in input/output devices, increasing the importance of input/output performance to overall system performance. Further, experience has shown that the performance of parallel input/output systems is particularly sensitive to data placement and data management policies, making good choices critical. To explore this vast design space, we have developed a user-level library, the Portable Parallel File System (*PPFS*), which supports rapid experimentation and exploration. The *PPFS* includes a rich application interface, allowing the application to advertise access patterns, control caching and prefetching, and even control data placement. *PPFS* is both extensible and portable, making possible a wide range of experiments on a broad variety of platforms and configurations. Our initial experiments, based on simple benchmarks and two application programs, show that tailoring policies to input/output access patterns yields significant performance benefits, often improving performance by nearly an order of magnitude.

1 Introduction

The widespread acceptance of massively parallel systems as the vehicle of choice for high-performance computing has produced a wide variety of machines and an even wider variety of potential input/output configurations, most with inadequate input/output capacity and performance. This disparity between computation and input/output rates reflects similar limitations on sequential systems. Processor performance has increased by several orders of magnitude in the last decade, but improvements in secondary storage access times have not kept pace (e.g., disk seek times have decreased by less than a factor of two even as data density has increased dramatically).

In an effort to balance impressive peak processing rates with sufficient input/output bandwidth, most parallel systems support multiple, redundant arrays of inexpensive disks (RAIDs) [21, 12]. Although multiple RAIDs provide *peak* input/output bandwidth equal to the product of the number of RAIDs and their individual bandwidths, little is known about the *effective* input/output bandwidth of an array of RAIDs in parallel scientific computing environments.

Developing models for how such massively parallel systems should be configured to be balanced, and how to manage the input/output resources to achieve high *sustained* performance requires exploration of a broad space of choices (e.g., disk data striping factors, file and disk block prefetch policies, and caching policies). Simply put, we must determine the most effective combinations of buffering, caching, data distribution and prefetching policies that allow a parallel file system to *reduce* the number of physical input/output operations and to *overlap* physical input/output with computation.

To explore these issues, we have developed a Portable Parallel File System (*PPFS*) to study the interaction of application access patterns, file caching and prefetching algorithms, and application file data distributions. *PPFS* consists of a user-level input/output library that is portable across both parallel systems and workstation clusters, permitting a wide range of experimentation with modest human resources. *PPFS* allows application control of a variety of file caching, prefetching, data layout, and coherence policies. This enables rapid exploration of a wide range of possibilities, facilitating rapid convergence on good policies for each file in the application. The *PPFS* application interface supports a number of predefined policies, but can also extend its repertoire by accepting user-defined layouts, access pattern declarations, and prefetching patterns. Interposing the libraries between the application and the system software has allowed us to more quickly experiment with a variety of data distribution and data management algorithms than would be possible via system software modifications.

In a preliminary study, we have measured the performance of *PPFS* on simple input/output benchmarks and on two input/output intensive programs, a gene sequencing matching application and a low temperature plasma electron scattering code. All of these experiments were conducted on a variety of Intel Paragon XP/S hardware configurations, using the Intel parallel file system

*Supported in part by the National Science Foundation under grant NSF ASC 92-12369, by the National Aeronautics and Space Administration under NASA Contracts NGT-51023, NAG-1-613, and USRA 5555-22 and by the Advanced Research Projects Agency under ARPA contracts DAVT63-91-C-0029 and DABT63-93-C-0040. Andrew Chien is supported in part by NSF Young Investigator Award CDA-94-01124.

Permission to make digital/hard copies of all or part of this material without fee is granted provided that the copies are not made or distributed for profit or commercial advantage, the ACM copyright/server notice, the title of the publication and its date appear, and notice is given that copyright is by permission of the Association for Computing Machinery, Inc. (ACM). To copy otherwise, to republish,to post on servers or to redistribute to lists, requires specific permission and/or fee.
ICS '95 Barcelona, Spain © 1995 ACM 0-89791-726-6/95/0007..$3.50

PDF
Help

You can view the full content in the following formats:

ICS

References

[1] Am~NDT, J. W. Parallel aenome Sequence Comparison Using an iPSC/2 with a Concurrent File System. Master's thesis, University of Hlluois at Urbana- Champaign, Department of Computer Science, Jan. 1991.
Google Scholar

[2] AYDT, R. A. SDDF: The Pablo Self-Describing Data Format. Tech. rep., University of minois at Urbana-Champaign, Department of Computer Science, Sept. 1993.
Google Scholar

[3] CxBrmRn, L.-F., AND LONG, D. D. E. Exploiting Multiple I/O Streams to Provide High Data-Rates. In Proceedings of the 1991 Summer Usenix Conference (1991), pp. 31-4s.
Google Scholar

[4] CHOUDHARY, A., BORDAWEKAR, R., HARRY, M., KRISHNAIYER, R., PONNUSAMY, R., SlNGH, T., AND THAKUR, R. PASSION: Parallel And Scalable Software for Input-Output. Tech. rep., Department of Electrical and Computer Engineering, Syracuse University, Nov. 1994.
Google Scholar

[5] CORBETT, P., AND FI~ITELSON, D. Design and implementation of the Vesta Parallel File System. In Sealable High-Performance Computing Conference (May 1994), pp. 63-70.
Crossref | Google Scholar

[6] ELFORD, C. L., KUSZMAUL, C., HUBER, J., AND MADHYASTHA, T. M. Design of a Portable Parallel File System. Tech. rep., University of Illinois at Urbana-Champaign, Department of Computer S Nov. 1993.
Google Scholar

[7] ELFORD, C. L., KUSZMAUL, C., HUBER, J., AND MADHYASTHA, T. M. Portable Parallel File System Detailed Design. Tech. rep., University of Illinois at Urbana-Champaign, Department of Computer Science, Nov. 1993.
Google Scholar

PDF
Help

Nov. 1993.
Google Scholar

[9] FRENCH, J. C., PRATT, T. W., AND DAS, M. Performance Measurement of the Concurrent File System

ICS

Digital Library | Google Scholar

[10] HARTMAN, J. H., AND OUSTERHOUT, J. K. Zebra: A Striped Network File System. In Proceedings of the Usenix File Systems Workshop (May 1992), pp. 71- 78.
Google Scholar

[11] HUBBtt, JR., J. V., ELFOKO, C. L., REED, D. A., CHmN, A. A., AND BLUMSNTHAL, D. S. PPFS: A High Performance Portable Parallel File System. Tech. Rep. UIUCDCS-R-95-1903, University of Illinois at Urbana Champaign, January 1995.
Google Scholar

[12] INTEL $UPERCOMPUTER SYSTEMg DIVISION. Paeagon XP/S Product Overview. Beaverton, Oregon, Nov. 1991.
Google Scholar

[13] KOTZ, D. Disk-directed I/O for MIMD Multiprocessots. Tech. rep., Department of Computer Science, Dartmouth College, July 1994.
Digital Library | Google Scholar

[14] KOTZ, D., AND ELLIS, C. $. Practical Prefetching Techniques for Parallel File Systems. In Proceedings of the First International Conference on Parallel and Distributed Information Systems (December 1991), pp. 182-189.
Digital Library | Google Scholar

[15] KOTZ, D., AND ELLIS, C. S. Caching and Writeback Policies in Parallel File Systems. Journal of Parallel and Distributed Computing 17, 1-2 (January and February 1993), 140-145.
Digital Library | Google Scholar

[16] KWAN, T. T., AND KEBD, D. A. Performance of the CM-5 Scalable File System. In Proceeding 199J A CM International Conference on Supercomputing (July 1994).
Digital Library | Google Scholar

[17] LoVsRso, S. J., ISMAN, M., NANOPOtmOS, A., NEsvmnw, W., Mm~, E. D., AND WHEELER, R. sf$: A Parallel File System for the CM-5. In Proceedings of the 1993 Summer Usenix Conference (1993), pp. 291- 305.

PDF
Help

Environments. In 199~ Scalable High Performance Computing Conference (May 1994), pp. 71-78.
Google Scholar

[19] MPI: MPI: A Message Passing Interface Standard. Tech. rep., Message Passing Interface Forum, May

ICS

Digital Library | Google Scholar

[20] NrBDLBMAN, S. B., AND WUNSCH, C. D. An Efficient Method Applicable to the Search for Similarities in the Amino Acid Sequences of Two Proteins. Journal of Molecular Biology 48 (1970), 444-453.
Google Scholar

[21] PATTBRSON, D., GmsoN, G., AND KATZ, R. A Case For Redundant Arrays of Inexpensive Disks (RAID). In Proceedings of A CM SIGMOD (December 1988), pp. 109-116.
Digital Library | Google Scholar

[22] PATTERSON, D., GmsoN, G., AND SATYANARAYANAN, M. A Status Report on Research in Transparent Informed Prefetcking. Tech. Rep. CMU-CS-93-113, Department of Computer Science, Carnegie-Mellon University, Feb. 1993.
Digital Library | Google Scholar

[23] REED, D. A. Performance Instrumentation Techniques for Parallel Systems. In Models and Techniques for Performance Evaluation of Computer and Communications Systems, L. Donatiello and R. Nelson, Eds. Springer-Verlag Lecture Notes in Computer Science, 1993.
Digital Library | Google Scholar

[24] REED, D. A. Experimental Performance Analysis of Parallel Systems: Techniques and Open Problems. In Proceedings of the 7th International Conference on Modelling Techniques and Tools for Computer Performance Evaluation (May 1994).
Digital Library | Google Scholar

[25] SHIBLDS, K., TAVSRA, L., SCLrLLIN, W. H., ELFORD, C. L., AND REED, D. A. Virtual Reality for Parallel Computer Systems Analysis. In A CM SIGGRAPH 'gg Visual Proceedings (July 1994), p. 261.
Google Scholar

[26] STELLNER, G., BODE, A., LAMBERTS, S., AND Luo- WIG, T. Developing Applications for Multi Systems on Workstation Clusters. In HPCN Europe, The International Conference and Exhi High- Performance Computing and Networking (1994).
Digital Library | Google Scholar

PDF
Help

Google Scholar

ICS

Cited By

View all

Zou Y, Chen C, Deng T, Zhang J, Zhu X, Chen S and Yin S. (2022). User-level parallel file system: Case studies and performance optimizations. Concurrency and Computation: Practice and Experience. 10.1002/cpe.6905. **34**:13. Online publication date: 28-Feb-2022.

https://doi.org/10.1002/cpe.6905

Chen C, Liu J, Zou Y, Deng T, Zhu X and Yin S. (2019). A Case Study on the Efficiency of User-Level Parallel File Systems. 2019 IEEE 21st International Conference on High Performance Computing and Communications; IEEE 17th International Conference on Smart City; IEEE 5th International Conference on Data Science and Systems (HPCC/SmartCity/DSS). 10.1109/HPCC/SmartCity/DSS.2019.00028. (90-97). Online publication date: Aug-2019.

https://doi.org/10.1109/HPCC/SmartCity/DSS.2019.00028

Garcia-Carballeira F, Calderon A, Carretero J, Fernandez J and Perez J. (2016). The Design of the Expand Parallel File System. The International Journal of High Performance Computing Applications. 10.1177/1094342003017001003. **17**:1. (21-37). Online publication date: 26-Jul-2016.

https://doi.org/10.1177/1094342003017001003

Show More Cited By

Index Terms

PPFS: a high performance portable parallel file system

PDF

Help

Computer systems organization

Computing methodologies

Distributed architectures

Simulation types and techniques

ICS

Grid computing

Massively parallel and high-performance simulations

Parallel architectures

Multicore architectures

Serial architectures

Superscalar architectures

Hardware

Software and its engineering

Communication hardware, interfaces and storage

Software organization and properties

External storage

Software system structures

Integrated circuits

Distributed systems organizing principles

Semiconductor memory

Grid computing

Non-volatile memory

PDF

Help

Recommendations

DAC '95: Proceedings of the 32nd annual ACM/IEEE Design Automation Conference

Read More

A multiple file write scheme for improving write performance of small files in Fast File System

ICS

Read More

Adding parity to the Linux ext3 file system

Modern disks no longer operate in a simple "fail-stop" manner, yet commodity operating systems assume they do. We design and implement a parity based approach to improve the robustness of journaling file systems. We modify the...

Read More

Comments

DL Comment Policy

Comments should be relevant to the contents of this article, (sign in required).

Got it

0 Comments

Share

Best Newest Oldest

Nothing in this discussion yet.

Privacy Do Not Sell My Data

PDF

Help

Download PDF

Categories

About

ICS

Magazines
Books
Proceedings
SIGs
Conferences
Collections
People

ACM Digital Library Board
Subscription Information
Author Guidelines
Using ACM Digital Library
All Holdings within the ACM Digital Library
ACM Computing Classification System
Accessibility Statement

Join

Join ACM
Join SIGs
Subscribe to Publications
Institutions and Libraries

Connect

Contact us via email
ACM on Facebook
ACM DL on X
ACM on Linkedin
Send Feedback
Submit a Bug Report

The ACM Digital Library is published by the Association for Computing Machinery. Copyright © 2025 ACM, Inc.

Terms of Usage | Privacy Policy | Code of Ethics

ACM DL DIGITAL LIBRARY
Association for Computing Machinery

PDF
Help