

The Design and
Implementation
of the **4.4 BSD**
Operating System

*Up and running.
Fully documented
by the system architects!*



Marshall Kirk McKusick

Keith Bostic

Michael J. Karels

John S. Quarterman

John Lasseter 94

This book is in the **Addison-Wesley UNIX and Open Systems Series**

Series Editors: *Marshall Kirk McKusick* and *John S. Quarterman*

Publishing Partner: *Peter S. Gordon*

Associate Editor: *Deborah R. Lafferty*

Associate Production Supervisor: *Patricia A. Oduor*

Marketing Manager: *Bob Donegan*

Senior Manufacturing Manager: *Roy E. Logan*

Cover Designer: *Barbara Atkinson*

Troff Macro Designer: *Jaap Akkerhuis*

Copy Editor: *Lyn Dupré*

Cover Art: *John Lasseter*

UNIX is a registered trademark of X/Open in the United States and other countries. Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and Addison-Wesley was aware of a trademark claim, the designations have been printed in initial caps or all caps.

The programs and applications presented in this book have been included for their instructional value. They have been tested with care, but are not guaranteed for any particular purpose. The publisher offers no warranties or representations, nor does it accept any liabilities with respect to the programs or applications.

Library of Congress Cataloging-in-Publication Data

The design and implementation of the 4.4BSD operating system /
Marshall Kirk McKusick ... [et al.].

p. cm.

Includes bibliographical references and index.

ISBN 0-201-54979-4

1. UNIX (Computer file) 2. Operating systems (Computers)

I. McKusick, Marshall Kirk.

QA76.76.063D4743 1996

96-2433

005.4'3--dc20

CIP

Copyright © 1996 by Addison-Wesley Longman, Inc.

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, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of the publisher. Printed in the United States of America. Published simultaneously in Canada.

Text printed on recycled and acid-free paper.

ISBN 0201549794

7 8 9 101112 MA 02 01 00 99*

7th Printing October 1999

Th
Withou
th

s into main memory
 memory when they are
 l to reside in virtual
 reside in main mem-
 e needed.

the kernel. 4.4BSD
 chine, repairing any
 on. See also *crash*

process as a result of

h an optional slash
 ated by slashes, and
 s with a slash, it is
 ins at the root direc-
 /*hostname*, and the path
 process. A slash by
 the current working

low on a connection.
 but the send window
 standing. If no win-
 v probe is sent.

h the system maps a

to as the *pmap* struc-
 on and access tables
 y-management hard-
 access rights, in addi-

ts the unidirectional
 stream-oriented, reli-
 with the "l" symbol.
 m a to the standard
 l | b".

tput of one process is

system to place pages

polling I/O The normal mode for a descriptor whereby the system will block if a read request has no data available or a write request has no buffering available. A process can determine whether an I/O operation will block by polling the kernel using the *select* system call. The *select* system call can be requested to return immediately with the information or to block until at least one of the requested I/O operations can be completed. See also *nonblocking I/O*; *signal-driven I/O*.

POSIX The standards group for P1003, the portable operating-system interfaces established by the IEEE. Its first established standard was the kernel interface, 1003.1, which was ratified in 1988.

prefetching The retrieval of data before they are needed. Many machines prefetch machine instructions so that they can overlap the time spent fetching instructions from memory with the time spent decoding instructions.

prepaging The prefetching of pages of memory. Prepaging is a technique used by virtual-memory systems to reduce the number of page faults.

probing The operation of checking to see whether a hardware device is present on a machine. Each different type of hardware device usually requires its own technique for probing.

process In operating systems, a task or thread of execution. In UNIX, user processes are created with the *fork* system call.

process control block (PCB) A data structure used to hold process context. The hardware-defined PCB contains the hardware portion of this context. The software PCB contains the software portion, and is located in memory immediately after the hardware PCB.

process group A collection of processes on a single machine that all have the same process-group identifier. The kernel uses this grouping to arbitrate among multiple jobs contending for the same terminal.

process-group identifier A positive integer used to identify uniquely each active process group in the system. Process-group identifiers are typically defined to be the PID of the process-group leader. Process-group identifiers are used by command interpreters in implementing job control, when the command interpreter is broadcasting signals with the *killpg* system call, and when the command interpreter is altering the scheduling priority of all processes in a process group with the *setpriority* system call.

process-group leader The process in a process group whose PID is used as the process-group identifier. This process is typically the first process in a pipeline.

process identifier (PID) A nonnegative integer used to identify uniquely each active process in the system.

process open-file table See *descriptor table*.

processor priority level A priority that the kernel uses to control the delivery of interrupts to the CPU. Most machines support multiple priority levels at which the processor may execute. Similarly, interrupts also occur at multiple