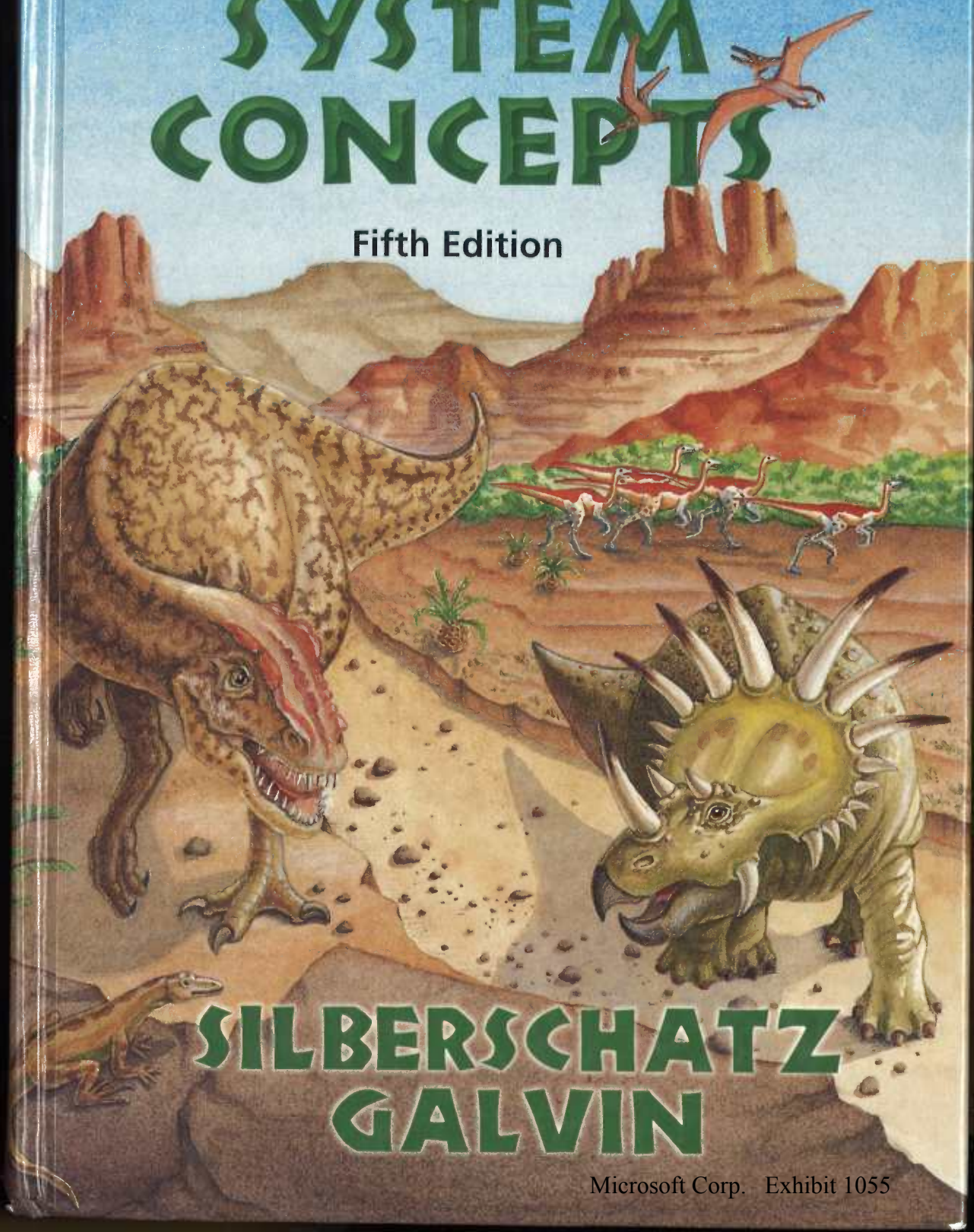
OPERATING SYSTEM CONCEPTS

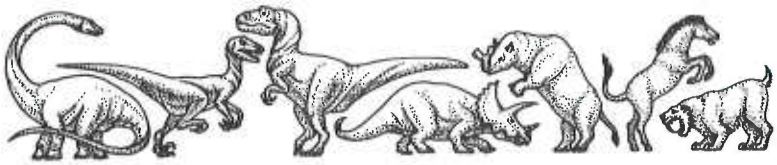
Fifth Edition



SILBERSCHATZ
GALVIN

OPERATING SYSTEM CONCEPTS

Fifth Edition



Abraham Silberschatz

Bell Labs

Peter Baer Galvin

Corporate Technologies, Inc.

ADDISON-WESLEY

An imprint of Addison Wesley Longman, Inc.

Reading, Massachusetts • Harlow, England • Menlo Park, California
Berkeley, California • Don Mills, Ontario • Sydney
Bonn • Amsterdam • Tokyo • Mexico City

Editor-In-Chief: *Lynne Doran Cote*
Acquisitions Editor: *Maite Suarez-Rivas*
Associate Editor: *Deborah Lafferty*
Production Editor: *Patricia A. O. Unubun*
Design Editor: *Alwyn R. Velásquez*
Manufacturing Coordinator: *Judy Sullivan*
Cover Illustration: *Susan Cyr*

Access the latest information about Addison-Wesley titles from our World Wide Web site: http://www.awl.com/cseng

Reprinted with corrections, February 1998.

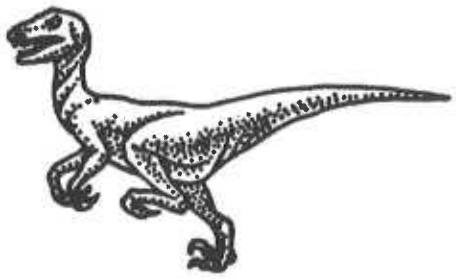
Library of Congress Cataloging-in-Publication Data

Silberschatz, Abraham.
Operating system concepts / Abraham Silberschatz, Peter Galvin. — 5th ed.
p. cm
Includes bibliographical references and index.
ISBN 0-201-59113-8
1. Operating systems (Computers) I. Galvin, Peter B. II. Title.
QA76.76.063S5583 1998

005.4, 3—dc21 97-28556
CIP

ISBN 0-201-59113-8
3 4 5 6 7 8 9 10 MA 01009998

Chapter 4



PROCESSES

Early computer systems allowed only one program to be executed at a time. This program had complete control of the system, and had access to all of the system's resources. Current-day computer systems allow multiple programs to be loaded into memory and to be executed concurrently. This evolution required firmer control and more compartmentalization of the various programs. These needs resulted in the notion of a *process*, which is a program in execution. A process is the unit of work in a modern time-sharing system.

The more complex the operating system, the more it is expected to do on behalf of its users. Although its main concern is the execution of user programs, it also needs to take care of various system tasks that are better left outside the kernel itself. A system therefore consists of a collection of processes: Operating-system processes executing system code, and user processes executing user code. All these processes can potentially execute concurrently, with the CPU (or CPUs) multiplexed among them. By switching the CPU between processes, the operating system can make the computer more productive.

4.1 ■ Process Concept

One hindrance to the discussion of operating systems is the question of what to call all the CPU activities. A batch system executes *jobs*, whereas a time-shared system has *user programs*, or *tasks*. Even on a single-user system, such as MS-DOS and Macintosh OS, a user may be able to run several programs at one time: one interactive and several batch programs. Even if the user can execute only one

program at a time, the operating system may need to support its own internal programmed activities, such as spooling. In many respects, all of these activities are similar, so we call all of them *processes*.

The terms *job* and *process* are used almost interchangeably in this text. Although we personally prefer the term *process*, much of operating-system theory and terminology was developed during a time when the major activity of operating systems was job processing. It would be misleading to avoid the use of commonly accepted terms that include the word *job* (such as job scheduling) simply because the term *process* has superseded it.

4.1.1 The Process

Informally, a *process* is a program in execution. The execution of a process must progress in a sequential fashion. That is, at any time, at most one instruction is executed on behalf of the process.

A process is more than the program code (sometimes known as the *text section*). It also includes the current activity, as represented by the value of the *program counter* and the contents of the processor's registers. A process generally also includes the process *stack*, containing temporary data (such as subroutine parameters, return addresses, and temporary variables), and a *data section* containing global variables.

We emphasize that a program by itself is not a process; a program is a *passive* entity, such as the contents of a file stored on disk, whereas a process is an *active* entity, with a program counter specifying the next instruction to execute and a set of associated resources.

Although two processes may be associated with the same program, they are nevertheless considered two separate execution sequences. For instance, several users may be running copies of the mail program, or the same user may invoke many copies of the editor program. Each of these is a separate process, and, although the text sections are equivalent, the data sections will vary. It is

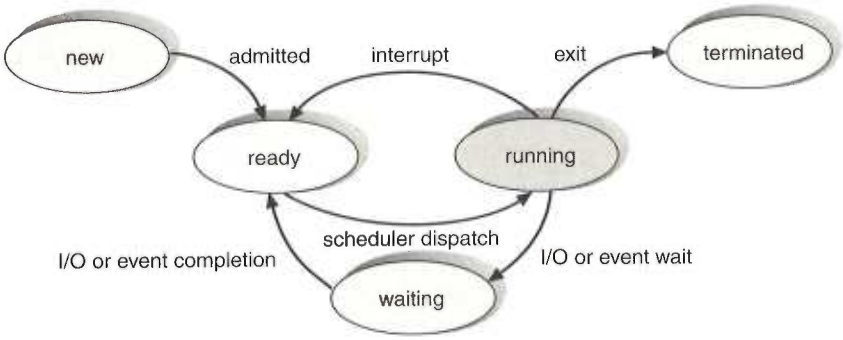


Figure 4.1 Diagram of process state.

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.