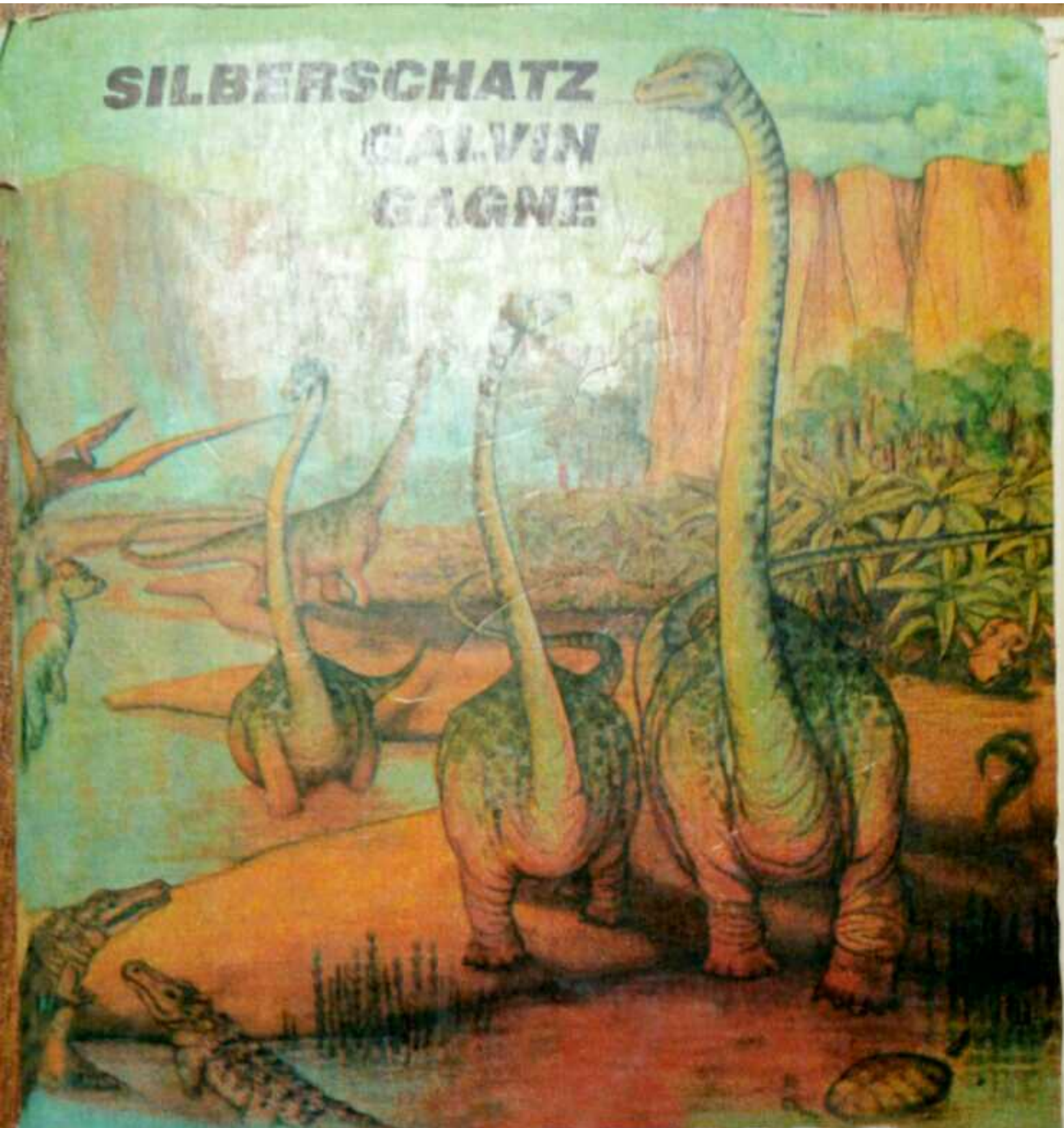
SILBERSCHATZ
GALVIN
GAGNE



OPERATING SYSTEM CONCEPTS

SIXTH EDITION

CONTENTS

PART ONE ■ OVERVIEW

Chapter 1 Introduction

Chapter 2 Computer-System Structures

Chapter 7 Process Synchronization

Chapter 8 Deadlocks

PART THREE ■ STORAGE MANAGEMENT

Chapter 9 Memory Management

Chapter 10 Virtual Memory

Chapter 11 File-System Interface

Chapter 12 File-System Implementation

PART FOUR ■ I/O SYSTEMS

Chapter 13 I/O Systems

Chapter 14 Mass-Storage Structure

PART FIVE ■ DISTRIBUTED SYSTEMS

Chapter 15 Distributed System Structures

Chapter 16 Distributed File Systems

Chapter 17 Distributed Coordination

PART SIX ■ PROTECTION AND SECURITY

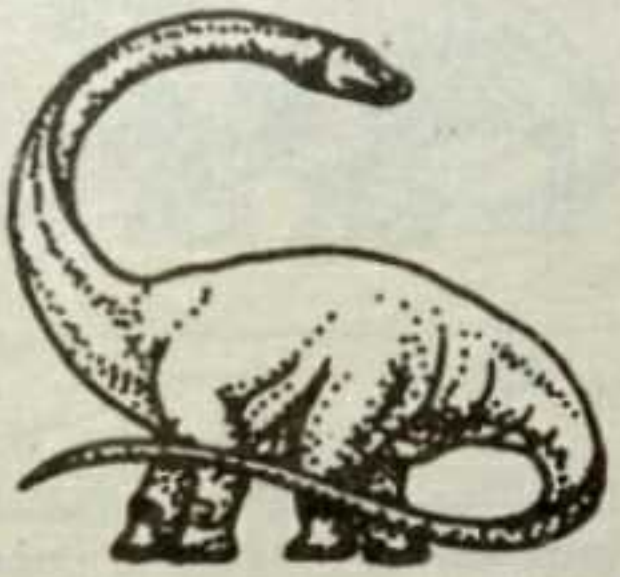
Chapter 18 Protection

Appendix A The FreeBSD System (contents online)

Appendix B The Mach System (contents online)

Appendix C The Nachos System (contents online)

Part One



OVERVIEW

An *operating system* is a program that acts as an intermediary between the user of a computer and the computer hardware. The purpose of an operating system is to provide an environment in which a user can execute programs in a *convenient* and *efficient* manner.

We trace the development of operating systems from the first hands-on systems, through multiprogrammed and time-shared systems, to current handheld and real- time systems. Understanding the evolution of operating systems gives us an appreciation for what an operating system does and how it does it.

The operating system must ensure the correct operation of the computer system. To prevent user programs from interfering with the proper operation of the system, the hardware must provide appropriate mechanisms. We describe the basic computer architecture that makes it possible to write a correct operating system.

The operating system provides certain services to programs and to the users of those programs in order to make their tasks easier. The services differ from one operating system to another, but we identify and explore some common classes of these services.