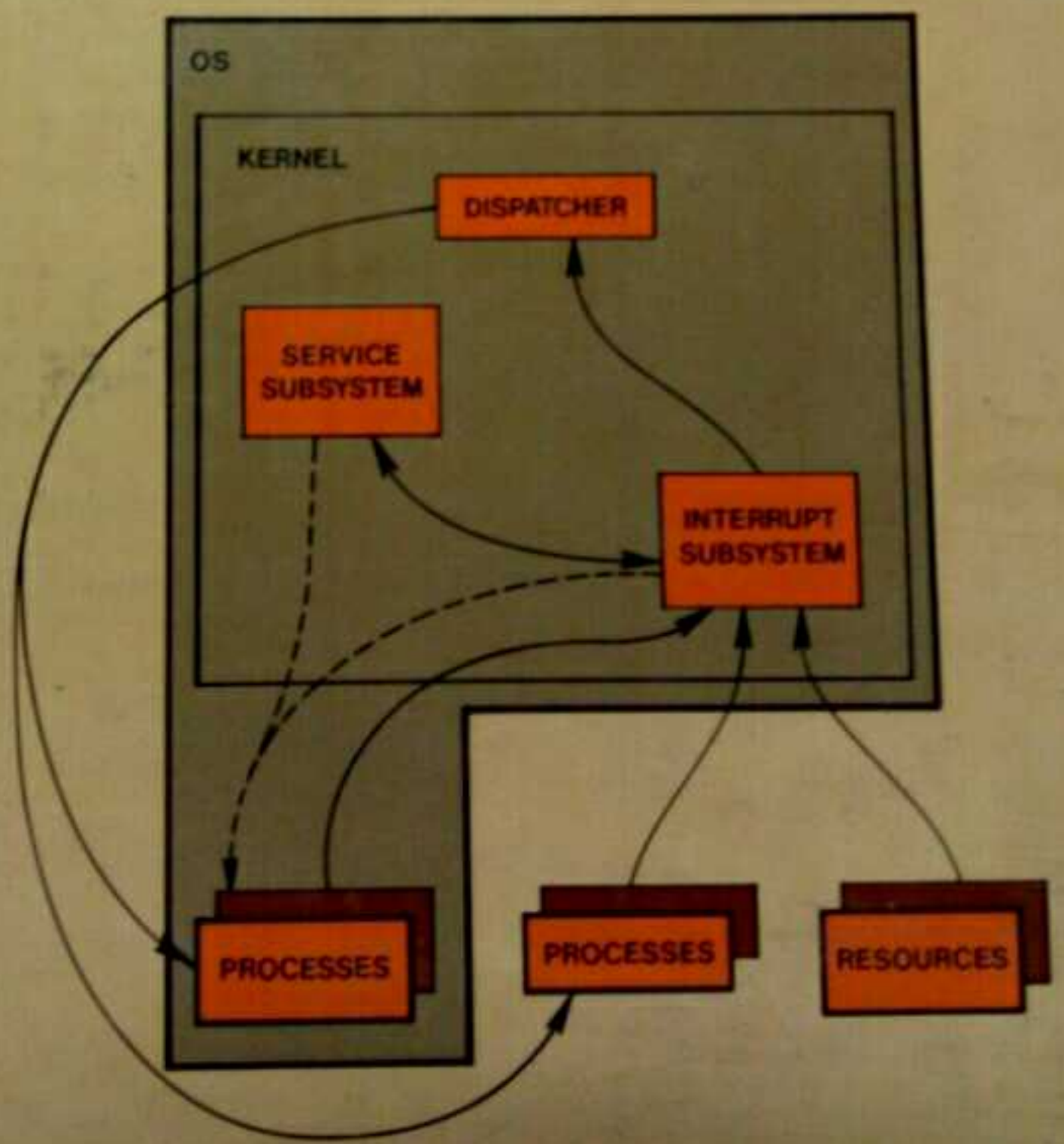
Raymond W. Turner

Operating Systems

Design and Implementations



Operating Systems

Design and Implementation

Raymond W. Turner
Senior Scientist, GHG Corporation
Houston, Texas

Macmillan Publishing Company
New York
Collier Macmillan Publishers
London

Contents

CHAPTER 1

Introduction

This is a book about operating systems and systems programming. What is systems programming? Some claim that it is programming done by "systems programmers" working for a computer manufacturer developing software delivered with the computers. If this definition is accepted, not only is operating system development encompassed by systems programming, but also the development of text editors, BASIC interpreters, COBOL compilers, linkage editors, scientific subroutine libraries, word processors, data base management systems, and so on. As a matter of fact, since most manufacturers have many standard industrial and commercial software applications packages available to enhance the marketability of their products, development of such programs as general ledger, inventory control, and energy management might also be considered systems programming.

Obviously, this definition is too broad to define topics to be covered adequately in a single-semester course. How about defining systems programming as the development of "nonapplications" programs? But what is an applications program, as opposed to a systems program? It is a program that executes under the control of the operating system, generally in a nonprivileged mode. But under this definition, all programs listed in the preceding paragraph, except the very heart of the operating system itself, could be considered applications. Should text editors and compilers be classified as applications programs? Systems programming might, therefore, relate to the programming of the operating system itself and nothing more. Thus it follows that the only "systems programmers" are the programmers who develop or maintain operating systems.

Now we have a definition that is probably too restrictive. For the purposes of this text, **systems programming** will be defined as programming that uses techniques similar to those commonly found in operating systems, even though the use may not be in direct support of the operating system. Generally, these techniques will be used to develop software that can provide a programming