COMPUTER INFORMATION SYSTEMS DEVELOPMENT: Design and Implementation

David R. Adams

Michael J. Powers

V. Arthur Owles

COMPUTER INFORMATION SYSTEMS DEVELOPMENT: Design and Implementation

David R. Adams
Associate Professor, Computer Information Systems
Northern Kentucky University
Highland Heights, Kentucky

Michael J. Powers
The Country Companies
Bloomington, Illinois

V. Arthur Owles
Industry Consultant
Wang Laboratories, Inc.
Lowell, Massachusetts

Published by
J86 **SOUTH-WESTERN PUBLISHING CO.**
CINCINNATI WEST CHICAGO, ILL. DALLAS PELHAM MANOR, N.Y. PALO ALTO, CALIF

ISBN: 0-538-10860-6

Library of Congress Catalog Card Number: 83-51621

2 3 4 5 6 7 8 K 7 6 5

Printed in the United States of America

PREFACE

PERSPECTIVE

This book, in part, represents an implementation of the *Model Curriculum for Undergraduate Computer Information Systems Education* of the Data Processing Management Association-Education Foundation (DPMA-EF). Specifically, the information presented in this book meets or exceeds the content called for in the suggested outline for course *CIS-5—Structured Systems Analysis and Design*. Correspondence between this book and the course specifications is assured by the fact that the text was developed under the oversight of the DPMA-EF, with content appropriateness and technical accuracy validated through independent review.

The DPMA-EF curriculum specifies a structured approach to systems development through use of structured methods within an established life cycle. The curriculum is aimed at graduating students qualified as entry-level programmer/analysts in business-oriented computer facilities.

CONTENT LEVEL

This text is designed to support an undergraduate course. It is assumed that students using this text will have completed an introductory course in computer information systems (CIS). Students should also have completed two semesters of work in structured programming designed to impart skills in the development of COBOL programs that solve business problems. Students should be familiar with the terms and techniques of program development to gain full value from a course based on this book.

In addition, this text is designed for use in a second course in systems analysis and design. Prior to use of this text, students should have completed a review of structured systems analysis work. A companion text, *Computer Information Systems Development: Analysis and Design*, by Powers, Adams, and Mills, is designed to provide a background leading up to the content of this text. However, the first two chapters of this book review the highlights of the companion text. Thus, students with sufficient background and qualifications could use this text without previous work in the field. Conversely, students who have just completed a course based on the companion work may be able to move quickly through the first two chapters to concentrate on the new material.

CONTENT HIGHLIGHTS

The book uses a basic, easily taught systems development life cycle as a framework. This life cycle divides a typical systems development project into five phases and 15 activities. This book concentrates upon the third phase of the model life cycle structure—Detailed Design and Implementation. As indicated, the first two phases are reviewed in the opening chapters. This review serves to establish a context for the main instructional content that follows. Finally, since emphasis is on design, the Installation and Review phases of the project structure are overviewed in the final chapters.

(The final phases of the project structure are encompassed in a companion course, CIS-7, entitled *Applied Software Development Project*. As the focal point of the CIS-7 course, students actually develop computerized information systems. A companion text, *Computer Information Systems Development: Management Principles and Case Study* is designed for use with this course.)

This book makes heavy use of case methods for illustration and instruction. Case citations and examples are provided at appropriate points throughout the text. In addition, an Appendix provides an opportunity for in-depth experience and systems development practice.

The Appendix reviews the principles and practice of systems development projects. Then, three separate case studies are provided as a basis for class or supplemental assignments.

The chapters of this book are divided into two categories:

- Phases and activities of the systems development life cycle.
- Skills applied in systems analysis and design.

'Activities' Chapters

These chapters deal with the individual activities of the Detailed Design and Implementation Phase of the project structure. For the other phases, activities are reviewed in single chapters. These activity chapters use standard subject headings and follow a common presentation pattern. Within each of these chapters, there are standard sections on:

- Activity Description
- Objectives
- Scope
- End Products
- The Process
- Personnel Involved
- Cumulative Project File.

For each activity, two of these areas are treated as keys to a student's understanding of the analysis process. They are the objectives of each activity and its end products. The other areas tend to be be natural consequences of these two.

'Skills' Chapters

The second series of chapters deals with the individual skills applied in analysis and design of computer information systems. These chapters cover:

- The Roots of Systems Design
- The Process of Systems Design
- The Technical Environment of Systems Design

- Application Design Strategies
- File and Database Design
- Foundations of Software Design
- Evaluating Software Design
- Software Design Strategies
- Test Specifications and Planning
- Software Testing Strategies

ACKNOWLEDGMENTS

To assure accuracy and appropriateness for the content of this text, a highly experienced, objective group of persons was asked to review the manuscript during development. The careful readings and thoughtful comments of this group represented, cumulatively, an important contribution to the soundness of this text. Their contributions are acknowledged with sincere thanks.

Donald R. Deutsch, General Electric Information Systems Co., Nashville, TN

Dr. W. Terry Hardgrave, George Mason University, Fairfax, VA

Dr. William Hetzel, Hetzel and Associates, Jacksonville, FL

CONTENTS

ESTABLISHING DESIGN SPECIFICATIONS I

Part I establishes a framework for the main topic of this text—systems development. It is assumed that, before enrolling in the course for which this text is used, you have had coursework in the prerequisite subjects of systems analysis and design. Because students may have used varying texts in prerequisite coursework, and also because there may have been a time interval between the study of analysis and design and the present course, this book begins with a review of the principles of analysis and design.

As the title for this part of the book implies, the end result of analysis and design phases of a systems project is a set of design specifications. These design specifications, in turn, become the basis for systems development and implementation.

This part has five chapters. The first chapter reviews the phases of the systems development life cycle leading up to the beginning of systems development. The remaining four chapters recap the specific skills applied and results derived from systems analysis and design.

If you have just completed your study of systems analysis and design—and if your instructor feels you are ready to move immediately into developmental topics—you may be instructed to begin your work in Part II of this book. If so, the content of Part I will still be valuable—possibly necessary—as reference material.

ACTIVITIES 1–6:

1 THE ROOTS OF SYSTEMS DESIGN

LEARNING OBJECTIVES

On completing reading and other learning assignments for this chapter, you should be able to:

- □ Explain the main differences between systems analysis and systems design.
- □ Give the rationale fðr adopting a structured methodology for development of computer information systems.
- □ Describe and list typical phases of the systems development life cycle (SDLC).
- □ Describe the transition from systems analysis phases of the SDLC to design phases.
- □ Explain how the documentation from systems analysis becomes input to systems design.
- □ Explain how software design differs from systems design.

BACKGROUND

This text is designed to support continuing study in computer information systems development. It is assumed that the student using this text has done prior work in the areas of identifying system needs, defining those needs, and developing user-oriented systems designs. The background and skills appropriate to this introductory-level work are covered in a companion text, *Computer Information Systems Development: Analysis and Design,* by Powers, Adams, and Mills.

The major portions of this text deal with the design and development that take place after nontechnical, user-oriented specifications have been developed. The specifications for what a system is to do are, in general, the products of *systems analysis. Systems design,* then, encompasses the formulation of the technical plans and methods to implement the specified system.

Establishing a Working Structure

The development of a computer information system is so large an undertaking that it is virtually impossible to understand all aspects of a problem at the outset. Some sort of structured approach is necessary. A proven, practical methodology is needed. A standard method for dealing with complexities or uncertainties lies in breaking jobs down into a series of individually definable activities. Although the overall system to be built may be complex and filled with uncertainties, the subparts are small enough to be grasped and managed separately. The structure that results establishes a series of orderly steps that organize and monitor the total effort—a *project.*

If an organization develops systems on a continuing basis, it is desirable to have standard project structures. Thus, systems development projects can be compared with one another. Further, people can be assigned to projects with the assurance that they will understand—and be able to complete—their assignments. For this reason, most computer information systems organizations establish a standard project structure, typically known as a *systems development life cycle (SDLC).*

THE SYSTEMS DEVELOPMENT LIFE CYCLE

A systems development life cycle, basically, is a formalized description of a project structure for the development of computer information systems. Any large organization that is involved continuously in the development of computer information systems will have established some kind of life cycle. The specific steps within any given life cycle will vary among organizations. However, the underlying principle remains the same: It is necessary both to establish a fundamental structure for the management of projects and also to apply structures that are comparable among projects. This comparability, in