information architecture / team review / tradeoff /
system structure chart / CASE / data dictionary / s
information center / data structure diagram / in
fourth-generation language / E-R model / fou
systems development life cycle / DBMS
function-oriented models / database /
islands of automation / feasibility /
data flow diagram / objectives / da
transaction processing system /
tangible benefits / control / t
end-user computing / FA
efficiency / database / e
factory automation / f
intangible benefits / TPS
decision support systems /
telecommunications / tactics /
office automation / requirements
physical model / MIS / team review
artificial intelligence / user interface /
expert system / EUC / logical model / ES
management information system / E-R mod
system structure chart / DSS / prototyping / syst
database management system / expert system / da
information architecture / islands of automation / D
transaction processing system / DFD / objectives / tra
urth-generation language / designer's tradeoff chart /

SYSTEMS DEVELOPMENT
A Practical Approach

William Amadio

SYSTEMS DEVELOPMENT
A Practical Approach

William Amadio
Rider College

SYSTEMS DEVELOPMENT
A Practical Approach

For Bill, Jr.—
you're the best!

PREFACE

To the Student

This text deals with the analysis, design, implementation, and maintenance of computer-based information systems in business organizations. These topics are of concern to *all* business professionals regardless of their field of specialization.

In reading this text, you will discover that the systems development process is made up of technical, economic, social, and political components. No one component may be considered to the exclusion of the others. If you are an information systems major, pay particular attention to the non-technical issues. If your major is outside of information systems, try to appreciate the importance of the technical issues to the process. When all parties recognize and understand competing points of view, a successful project results.

I am happy you decided to take this course; I hope you find it interesting and worthwhile. Good luck!

To the Instructor

Every instructor who regularly teaches systems analysis and design knows the meaning of the phrase "love-hate relationship." We know the importance, the beauty, and the excitement of the subject, but we also know the difficulty of teaching it and the frustration that comes when our results do not match our expectations.

I believe the source of our difficulty is articulated in Kolb's Experiential Learning Model, which states that the learning/problem solving process is "both active and passive, concrete and abstract. It [the process] can be conceived of as a four-stage cycle: (1) concrete experience is followed by (2) observation and reflection which leads to (3) the formation of abstract concepts and generalizations which lead to (4) hypotheses to be tested in future action which in turn leads to new experiences."[1]

[1]D. Kolb, I. Rubin, and J. McIntyre. *Organizational Psychology* (Englewood Cliffs, NJ: Prentice-Hall, 1974), p. 28.

The difficulty of teaching the systems course lies in the fact that most undergraduate students do not bring concrete systems experience to the course; they usually bring programming experience. The stage hypothesis of Kolb's model would suggest, then, that the systems instructor's first task is to expand the students' programming experience into concrete systems experience. Otherwise, the instructor is beginning the learning cycle at stage 2, which translates into trying to teach students to analyze, design, implement, and maintain something they have never seen.

I believe that the combination of microcomputer technology, fourth-generation languages (4GLs), and computer-aided software engineering (CASE) tools provides the means to implement the full learning cycle in the systems course. This technology can be exploited to create a new learning environment in which students acquire task-specific and critical thinking skills that are transferable to any hardware/software work environment.

Two complete, microcomputer-based systems accompany this text. One of them provides a meaningful first experience with an information system operating within the context of a business organization. With this common base of experience, the classroom becomes a place for students and instructor to observe, reflect, conceptualize, and generalize together. The learning cycle is complete when the second system is used to provide an arena for active experimentation with the newly developed concepts and skills.

This *do it—study it—do it again* approach is fundamental to the design of this text.

Audience

Systems Development: A Practical Approach is designed for a first course in systems development. This course is usually offered at the sophomore and junior levels by both two-year and four-year institutions.

Organization

The text is organized into five parts. Part One provides an overview of systems development practice and introduces the Horatio & Co. Cost Control System, a working system provided on diskette with this text. The software development environment is dBASE III PLUS supplemented with GENIFER, a CASE tool that provides a data dictionary, report writer, screen painter, and dBASE code generator.

GENIFER proved to be just the right tool to bring a working system into the classroom without getting bogged down in programming details.

GENIFER, dBASE, and cost control system screens are incorporated directly into the text to support a *generic* discussion of system concepts. **Student and instructor familiarity with dBASE III PLUS and GENIFER is not assumed or necessary.**

Parts Two through Four of the text look back in time to reconstruct the development of the cost control system from problem analysis through integration of the system with the existing infrastructure of the organization.

Part Five examines technical and organizational issues not covered in the development of the cost control system and provides a look at trends that are likely to affect systems development practice in the future.

After an overview of systems development practice is presented in Chapter 1, the Kolb four-stage model is incorporated into the structure of the remaining chapters of Parts One through Four. Each chapter begins with a concrete experience with the Horatio & Co. Cost Control System. These experiences include using the software, reconstructing analysis and design sessions, preparing deliverables, and reviewing decisions.

The latter part of each chapter reflects on the new experience and generalizes to the appropriate concepts. Each chapter concludes with a set of thought questions designed to stimulate further observation, reflection, and conceptualization and a set of exercises that asks the student to apply the newly learned concepts to the development of the Horatio & Co. Credit Union System, a second working system supplied on diskette with this text. The *Instructor's Manual* accompanying the text contains complete answers to all 78 thought questions and 94 exercises.

In making choices about conceptual content, I relied on the model curricula of DPMA and ACM and the work of Vitalari.[2] Any shortcomings, of course, rest with me.

More often than not, the difficult choices involved not *what* to cover but rather *when* to cover it. Because of the close connection to a real system, topics like feasibility, organizational structure, management approval, and effective communication made their presence felt in each and every chapter.

To help with your assessment of the content, I have prepared the accompanying analysis, which lists the dominant theme(s) of each chapter, and representative figures that communicate the flavor of some of the treatments.

---

[2]See N. Vitalari, "Knowledge as a Basis for Expertise in Systems Analysis: An Empirical Study," *MIS Quarterly* 9, (September, 1985): 221–40.

Dominant Chapter Themes	*Representative Figures*
Part One: The Foundation of This Text	
1. Overview	1.2, 1.3
2. System outputs: reports and inquiries	2.6, 2.13
3. System inputs: maintenance processes	3.4, 3.5, 3.16, 3.17
Part Two: Systems Analysis	
4. Objectives Tactics Information system functions	4.2, 4.4
5. DFD-based modeling Requirements Communication techniques	5.11, 5.23, 5.25, 5.26
6. Assessing benefits Boundaries: system wide automated/human work mix	6.1, 6.6, 6.11
7. Feasibility: operational technical economic Buying software Building software: prototyping detailed design	7.1, 7.4, 7.5, 7.6
Part Three: Systems Design	
8. Management approval Project management Design: hardware procedures personnel	8.2, 8.3, 8.4
9. Design: data software	9.13, 9.26, 9.28, 9.30
10. Inside a CASE-developed system	10.4, 10.7, 10.8, 10.14
11. Evaluation: refining prototypes team reviews	11.1, 11.8
Part Four: Completing the Systems Development Life Cycle	
12. Implementation and integration	12.1, 12.5, 12.6, 12.8
13. Ongoing documentation	13.1
14. A critical review of the Horatio systems	14.1, 14.2

Dominant Chapter Themes	*Representative Figures*
Part Five: Systems Development Practice—Today and Tomorrow	
15. A critical review of the technology	14.1
16. More organizational issues	16.1
17. Bigger and better CASE	17.2
Data-oriented modeling	17.5

Highlights

Several features of this text deserve special mention.

▶ **New learning environment.** This text presents an organized set of learning experiences built around two microcomputer-based application systems that are supplied on diskette. The text first introduces the students to the systems as users. Subsequently, they learn the principles of analysis and design through a reconstruction of the development of these systems. Every phase of the life cycle is relevant because the students have seen and used the finished product and can evaluate development activities in terms of their own experience.

▶ **Active use of CASE technology.** The Horatio & Co. Cost Control System was developed with dBASE III PLUS supplemented by the GENIFER CASE tool. In Chapter 10, students use GENIFER's documentation database to learn the inner workings of the cost control system in the same way they would use any CASE design database in practice. The end-of-chapter exercises ask students to construct CASE specifications for the credit union system. Thus, there is no need to "cover" CASE technology. Students use it actively in their pursuit of the text material.

▶ **Expanded systems development life cycle.** Traditional evaluation of alternatives is expanded to include choice of software development methodology. Prewritten software packages, prototyping, and detailed design are presented as a portfolio of methods to be used according to the circumstances of the project. The use of each method is integrated into the unifying framework of the expanded life cycle.

▶ **Full spectrum of information technology.** The Horatio & Co. systems cover a range of applications from transaction processing through decision support. From the very outset, these systems are discussed within a broad context of information technology that includes expert systems, office automation, telecommunications, and factory automation. Part

Five discusses the dominant issues faced in the analysis, design, and implementation of these newer technologies.

▶ **Higher-level conceptual treatment.** The specific-to-general approach that dominates the text is exploited several times to reach conceptual levels not normally associated with this course. For example, using the Horatio & Co. organization as a base, Part Five reviews the McKenney & McFarlan model of technology assimilation and discusses its implications for systems analysis and design. The concrete experiences provided in this text place within reach organizational concepts normally covered in higher-level undergraduate and graduate courses.

Instructor's Support Materials

This text is accompanied by four diskettes—an educational version of dBASE III PLUS (two diskettes), the Horatio & Co. Cost Control System diskette, and the Horatio & Co. Credit Union System diskette—plus an *Instructor's Manual.*

Both applications are self-contained, menu-driven systems. The cost control system runs under dBASE; the credit union system programs have been compiled, so they do not need dBASE to run. Both systems run in a dual-floppy or hard disk environment. The cost control system needs 384K; the credit union system runs in 256K RAM.

No knowledge of dBASE is assumed, and if you choose, no class time need be spent on the dBASE environment. The discussion of ad hoc reporting in the text is generic and can be applied to any interactive database environment. If you prefer, the dBASE Assistant used as an example of a graphic user interface can be replaced with any other graphic environment. If you want to do more with dBASE, the *Additional Materials* section in Chapter 9 of the *Instructor's Manual* contains material that I have found useful.

I began teaching systems analysis and design in 1979. By 1982, I had abandoned the course in frustration. In 1986, I returned to the course, armed with a set of dBASE exercises that eventually grew into the Horatio & Co. systems. All of the material in the text and *Instructor's Manual* has been refined through classroom testing.

The *Instructor's Manual* is an integral part of the course delivery package. The main text was written for the students; its purpose is to provide

concrete experience and to guide observation and conceptualization activities at the students' level. In the *Instructor's Manual*, however, we talk about broader contexts, supplementary issues, motivation, and evaluation.

Each chapter of the *Instructor's Manual* corresponds to one chapter in the text. Each *Instructor's Manual* chapter is composed of nine sections:

    I. Chapter Objectives

   II. Glossary Terms Introduced in This Chapter (170 in all)

  III. Teaching Tips and Strategies

  IV. Transparency Masters

   V. Topic Outline

  VI. Additional Materials

 VII. Answers and Guidelines for Thought Questions (78 in all)

VIII. Answers and Guidelines for Exercises (94 in all)

  IX. Test Items (30 per chapter—510 in all)

Sections III, VI, VII, and VIII deserve special mention because they support the unique approach used in the text. Because of its concrete and specific nature, students can master the text material on their own. You will find that only a portion of class time need be devoted to discussing the material the students read in the text. The remainder can be spent expanding observations, making more general conceptualizations, and experimenting with the newly learned concepts.

Part III of each chapter presents a plan for carrying out these activities. Part VII provides complete answers to each thought question along with guidelines on using the question to observe and conceptualize. Part VI provides additional relevant materials not presented in the text that can be used to broaden observations and generalize concepts as needed. This section contains *complete* treatments, not just citations and additional references. Finally, Part VIII provides *complete* answers to each exercise along with guidelines on using the exercise to test ideas and generate new experiences.

If you are new to systems courses, the package of text, software, and instructor's manual provides a comprehensive set of materials to guide you through your first offering. As you experience, observe, conceptualize, and experiment with the course, you will tailor the materials to suit your own environment, interests, and style.

If you are an experienced systems teacher, you probably have built your own collection of techniques and materials. Since the topics covered in the text are all "mainstream," substitution and experimentation should be quite simple. Good luck!

Acknowledgments

Many individuals contributed their expertise to the development of this text. I am especially indebted to Karen Calkins of Bryant College; Karen-Ann Kievit of California State University, Long Beach; Anne McClanahan of Ohio University; and John Stoob of Humboldt State University. They need only compare their reviews of earlier drafts with the final product to determine the extent of their contribution.

Special thanks must go to Lisa Gilmour of Ashton-Tate and to Jim Porzak of Bytel Corporation for their vision and enthusiasm and to the trustees of Rider College for granting a research leave to complete this project.

I also wish to thank the supporters and developers of the 1987 AACSB Advanced MIS Faculty Development Institute at Indiana University, especially those involved in the systems area: Ken Kozar, Milt Jenkins, Bob Bostrom, and Nick Vitalari.

Thanks go also to the following people who provided reviews, valuable comments, and other assistance:

Brother William Batt
Manhattan College

Cynthia Mathis Beath
University of Minnesota

Paul Borkowski
Cuyahoga Community College

Patrick Fenton
West Valley College

C. Brian Honess
University of South Carolina

Robert Keim
Arizona State University

Constance Knapp
Pace University

David Kroenke

Richard Leifer
Rensselaer Polytechnic Institute

Charles MaWhinney
Bentley College

Richard Meyer
Hartnell College

R. J. Planisek
Grand Valley State College

Steven Seilheimer
Niagara University

Charles Silcox
Widener University

Ronald Thorn
Grambling State University

James Walters
Pikes Peak Community College

Ronald Williams
Central Piedmont Community College

James Wilson
Triton College

Participants, 1988 National Computer Educators Institute
James Madison University

Friends and colleagues at Mitchell Publishing, Inc.

In every team project, there is one person at the center who sets the tone, leads by example, and "makes it all happen." This project is no exception. Erika Berg, editor extraordinaire, did all those things and more. I am truly grateful for the opportunity to know her and to work with her on this project.

At this point, many authors thank their spouses for tolerating long periods of separation and solitude. Fortunately, I do not have to do this. My wife, Camille, compiled detailed class notes from my analysis and design course in 1986 and again in 1987. She read and reread the manuscript, often discovering gaping holes in the development. She provided a second pair of eyes and ears during the feedback process. She kept me laughing. . . . Thanks, Camille.

Bill Amadio

CONTENTS

PART THREE
Systems Designs   174