Using Applications Software

An Introduction Featuring Framework

By Donald H. Beil

WORD PROCESSING SPREADSHEETS BUSINESS GRAPHICS DATABASE

Using Applications Software

An Introduction Featuring Framework

By Donald H. Beil

ISBN 0-07-079651-3

Published by Ashton-Tate Publishing Group
20101 Hamilton Avenue
Torrance, California 90502-1319

The software, computer, and product names mentioned in *Using Applications Software: An Introduction Featuring Framework™* are manufacturer and publisher trademarks and are used only for the purpose of identification.

Disclaimer: **This book and accompanying magnetic disk are designed to help you improve your computer use. However, the authors and publishers assume no responsibility whatsoever for the uses made of this material or for decisions based on its use, and make no warranties, either express or implied, regarding the contents of this book or any accompanying magnetic disks, its merchantability, or its fitness for any particular purpose.**

Neither the publishers nor anyone else who has been involved in the creation, production, or delivery of this product shall be liable for any direct, incidental, or consequential damages, such as, but not limited to, loss of anticipated profits or benefits resulting from its use or from any breach of any warranty. Some states do not allow the exclusion or limitation of direct, incidental, or consequential damages, so the above limitation may not apply to you. No dealer, company, or person is authorized to alter this disclaimer. Any representation to the contrary will not bind the publisher or the author.

10 9 8 7 6 5 4 3 2
Senior Editor: Brenda Johnson-Grau
Editor: James S. Bradbury
Production Editor: Kathryn M. Schmidt
Assistant Editor: Shaista P. Ali
Technical Editor: Ralph Davis
Design Director: Thomas Clark
Text Design: B. T. Miyake
Illustrations: Marian Haley Beil
Cover Art: Carol M. Barth
Cover Photography: Mark Heliger

Dedication

For Gabe and Noah

Acknowledgments

It's a pleasure to recognize others who have contributed to this book.

NTID/RIT

I acknowledge the assistance which I received from my employer, the National Technical Institute for the Deaf at the Rochester Institute of Technology (NTID/RIT). The RIT policy in support of sabbaticals for employees allowed me to concentrate on the completion of this manuscript. In particular, I want to recognize: Thomas Plough, RIT Provost and Vice President of Academic Affairs; Peter Pere, then Dean of NTID; Chuck Layne, then Assistant Dean, NTID Business Careers Division; and Robert Berl, Chairperson, NTID Data Processing Department. Each recognized the importance of this project to me and provided me with encouragement and support.

Barry Culhane graciously provided details of the organization of student activities at RIT—information that became the core of the examples in the book. Others at NTID/RIT who were encouraging or who provided help include John Sweeney, Chris Licata, Bruce Peterson, Ed Maruggi, Paul Taylor, and M. Richard Rose, President, RIT.

Ashton-Tate

Larry Benincasa, an old friend and presently a vice president at Ashton-Tate, was instrumental in early discussions of this book; for the last six years he and I have been discussing what we felt was the need for this text.

Bill Jordan, Robert Hoffman, Brenda Johnson-Grau, James Bradbury, Tom Clark, B.T. Miyake, and Larry Colker provided editorial and production assistance with the book. Ralph Davis verified the technical accuracy of the manuscript. David Helms was helpful in providing details of the limited-use version of Framework.

McGraw-Hill Book Company

Gale K. Murray, Representative, College Division provided early, valued help. Christina Mediate provided important editorial contributions, and Debbie Dennis and many others continue to provide marketing services.

Reviewers

I greatly appreciate reviews of the book in manuscript form prepared by: Robert E. Norton, Chairman, Computer and Information Sciences Department, San Diego Mesa College; Dr. Anthony Verstraete, The Pennsylvania State University; and Donald L. Davis, The University of Mississippi. I also appreciate comments received from Tapie Rohm at California State University—San Bernardino.

Others

Joseph E Beil, my father, again provided Wall Street Journal clipping service. Frank E. Cable, of The Pennsylvania State University, unknowingly influenced the first chapter, and provided other very important help. I discussed the project with Dan McCracken, and with Janice and Brian Wormington, and received encouragement from them.

Dom Fantauzzo of Logical Operations, Rochester, New York, as always, provided expert technical advice when I needed it.

I learned from a book entitled Student Instructions for Special Limited Version of Framework, by Thomas F. Goldman, published by McGraw-Hill Book Company, 1985.

And finally, my thanks to my wife Marian, who created the art for the book, and to our sons Noah and Gabe.

PREFACE

I'm excited about this book because I've long felt that there is a need for introductory computing courses that present problem solving with personal computers. This book provides that introduction by discussing the most popular personal-computer tools, often called applications software. These tools include word processing, spreadsheets, idea processing, business graphics, database, and integrated software.

My intention is that readers use this book and its tutorial approach to develop computer skills useful in many environments — in school, on the job, or at home. The skills you'll develop are among those most widely acquired by personal computer users. I hope that they are skills that will serve you throughout your career.

The book is unique in its presentation of this material, for each tool is discussed from two perspectives.

- First, applications software is presented generically; that is, the book provides an introduction to the general categories of software available in the marketplace.
- Second, material is presented with a hands-on tutorial approach that uses the limited-use version of Framework included with the book.

By studying and using this book, you'll be able to learn about applications software with material specifically developed for an academic environment, complete hands-on assignments, work on a realistic case study that parallels the presentation of content, have ready access to integrated software, and learn to use a real product.

Don Beil
Spring, 1986

Other books by Donald H. Beil:

Dynamics of Jazz,™ 1986
Symphony™ First, 1985
The Bank Street Writer™ Book, 1985
Using the Horizon™ Spreadsheet: with the UNIX™ Operating System, 1984
The DIF™ File: For Users of VisiCalc® and other Software, 1983
SuperCalc®! The Book, 1983
The VisiCalc® Book for the IBM® Personal Computer, 1983
The VisiCalc® Book, ATARI® Edition, 1983
The VisiCalc® Book, Apple® Edition, 1982
File Processing with COBOL, 1981

TABLE OF CONTENTS

CHAPTER 1 OVERVIEW

personal computers
applications software

In this book, we'll concentrate on the use of problem-solving tools with **personal computers**. We'll provide an introduction to the most widely used personal-computer tools, often called **applications software**. These include word processing, spreadsheets, idea processing, database, business graphics, and integrated products.

WHAT'S IN THIS BOOK?

integrated software

We'll explain these terms in the next chapter and discuss their use throughout the remainder of the book. The tools are presented from two perspectives. *First*, each tool is presented generically; that is, we'll discuss the capabilities and limitations provided by each category of tool. *Second*, we'll provide an opportunity to learn about these tools through a **hands-on** approach that uses the Framework® software included in this book. Framework is an **integrated software** product (published by Ashton-Tate®, a software developer and publishing company). *Integrated* means that this single product combines multiple functions.

WHAT'S NOT IN THIS BOOK?

A number of introductory computing topics are not covered in this book, for example, binary numbers, programming, and the history of computing are not included. Because these topics are of interest and importance for many computer users, it is fortunate that a large number of other books and courses in the computer science area provide this information.

GOALS OF THE BOOK

The course is designed to help you develop a selected set of computer skills useful in the real world, whether that world is school, a job, or elsewhere. I have selected this set of skills from the broad variety of computer skills an individual can acquire and I have restricted these to the introductory skills associated with using software in the categories listed above.

Word processing, for example, is a powerful general-purpose computer-based writing tool that's useful in English Composition and other courses that require written work. The word-processing knowledge and skill gained in the classroom is also useful on the job for written communication. Finally, in addition to being useful in school and on the job, many individuals use word processing in their personal correspondence.

The set of selected skills presented here are among those most widely acquired by personal-computer users. It's the great number of users of software in these categories who account for the rapid growth in the popularity of personal computers during recent years. A very large proportion of the personal computers in the workplace and at home are in use with software in the categories we'll discuss in this book. In general terms, these are our goals:

- Master the *concepts* associated with using personal computer applications software in the categories listed above.
- Develop *hands-on skill* with the major categories of software available for personal computers.
- Develop an appreciation of the fact that more than one type of computer tool may be needed to solve a problem and understand when and how multiple tools can be used together for *problem solving*.
- Understand the *capabilities* and *limitations* of each type of tool.
- Develop the ability to recognize problems which can be solved by personal computer applications software and then select an *appropriate tool* for the job at hand.
- Recognize problems which cannot be solved by these tools and distinguish between problems which have *computer solutions* with computer resources other than applications software and those which *do not* have computer solutions.
- Develop an understanding of *basic computer terminology*.

WHY FRAMEWORK?

hands-on skills

Framework was chosen for several reasons. *First*, it has capabilities in each of the areas we intend to study. This means that by learning to use it, we can achieve our first two objectives: mastering the *concepts* of personal computer applications software and developing *hands-on skills* with personal computers. Because the product encompasses a variety of tools, we can also discuss how those capabilities can be used together for problem solving, our third goal. We can study the concepts typically associated with integrated products — those products which contain several different problem-solving tools.

Second, Framework is a single product and therefore has learning advantages for those who are new to computers. Because it's a single product, what's learned in one area of functionality often transfers to another. There's the implication that when we've mastered one of the tools of the product, the others will be easier to learn. In an academic setting, using a single product with multiple capabilities is preferable to having to discuss and learn several software products, each of which performs one function.

New users may not realize how much learning energy is required to become comfortable with most computer products. Experienced users know that different products are usually built upon unique designs that must be absorbed and understood in order to use the product. For new users, learning to use a single general-purpose product is probably simpler than having to learn a series of separate tools that together provide the same capability.

However, there's a sacrifice here also, for concentrating on one product for the hands-on portion of the learning means that we will not study the wide variety of approaches used by different software producers to provide similar capabilities. Also, many personal-computer users believe that single integrated products that try to do everything succeed only by sacrificing depth within one or more of their separate tools.

There's another reason for using Framework. For the purpose of this and other books, Ashton-Tate has prepared a special limited-use version of Framework. It contains the bulk of the functionality of the original product, but is limited in other ways. For example, the unrestricted, commercially available version of Framework (which costs hundreds of dollars per copy) allows users to write lengthy documents with its word processor, whereas the limited-use version severely restricts document length. (See Appendix A.)

From our point of view, the restriction on document length will not hamper instruction on word-processing concepts. From Ashton-Tate's point of view, the limited-use version will not harm sales of its unrestricted product, given that the limited-use version has little value to someone in a business environment. Thus, in creating the limited-use version, the majority of the features have been preserved — without destroying the commercial value of the full-featured product.

Because the limited-use version is included on the diskette enclosed with this book, *a copy of the program is readily available to each student with the book*. In the past, software producers have focused on sales of their prod-

ucts at regular price to their commercial customers and, with a few exceptions, have not taken steps to make software available for use by students at lower costs. The approach taken by Ashton-Tate provides a vehicle for the academic study of personal-computing software without destroying the place that the full-featured product enjoys in the business community. The availability of this limited-use version was a major reason for the selection of Framework for this text.

AUTHOR AND READER CAUTIONS

In many ways this is a unique, positive position for an author. I'm able to talk generally about categories of software that are of great interest to me and I can present hands-on experiences based on a specific product that is readily available to every reader.

You as the reader and I as the author both know that the book and the software have been published by the same company and that the publishers hope you will use their products in the future. From my perspective this places a unique responsibility on me as an author. In the hands-on sections of the book that discuss Framework, I've tried to provide an honest analysis of the product by writing about it with integrity, presenting its capabilities as well as its limitations. You can be the judge as you work with the the book and product.

I like using Framework and I believe it's an innovative product in the personal-computer market. It has also been the winner in product comparisons reported by computer magazines. For example, *PC Magazine* sponsored a product face-off in which 29 MBA candidates formed teams to compare ten different products in the solution of a series of business problems. The Framework team was judged the winner. (*PC Magazine*, May 28, 1985). I did not, however, use Framework or an IBM personal computer as my word-processing tools in preparing the book. From habit and preference, I used a word processor on a different brand of computer.

I've tried throughout the book to present the foundation topics for each tool, the topics that are common to most product implementations of the tool. It is my hope that this book and this approach will help you build a foundation that's useful no matter what computer system you may use in other computing activity.

SOME PERSONAL NOTES

As a writer, my most important goal is to write with respect for the reader. For me this means assuming an intelligent reader who enjoys learning. Because each of you has a different learning style, I try to provide a variety of writing techniques, artwork, problems, and assignments.

I avoid telling you that anything is "easy." Using a personal computer requires skill, knowledge, and experience. We need time, effort, and patience to use personal computer software tools effectively.

I'm a teacher of the deaf at the National Technical Institute for the Deaf at the Rochester Institute of Technology (NTID/RIT) where I've been

teaching Computer Science to deaf college students for the past eleven years. Before joining NTID/RIT, I taught Computer Science at The Pennsylvania State University for four years. I also worked for the Peace Corps for five years: I taught mathematics in the Somali Republic, East Africa for two years and I worked in the Washington, D.C., office as a *systems analyst* for three years.

Personal computer software has fascinated me since 1978. My principal interests are in the use of computers in the area of productivity — the subject of this book — and in education. This is my eleventh book about computers and my second textbook (the first is an advanced book that discusses computer programming with the COBOL language).

I own seven personal computers with all kinds of peripheral equipment (printers, plotters, and so forth) and software, and I use computers for long hours every day — mostly for writing, but also because of the pleasure of using the technology.

If you want to contact me, here is my address:

National Technical Institute for the Deaf
Rochester Institute of Technology, Building 14
One Lomb Memorial Drive
Post Office Box 9887
Rochester, NY 14623-0887

I answer my mail.

CHAPTER 1 VOCABULARY

My primary concern is that you develop an understanding of the *meaning* of computing terms, rather than an ability to *define* the words. In the vocabulary lists at the end of each chapter, the terms are divided into several categories including general-purpose computing terms, terms specific to IBM personal computers, terms specific to Framework, and other terms.

General Computing Terms

Applications software
Binary numbers
Business graphics
Computer
Database
Hands-on
Idea processing
Integrated software
Peripheral equipment
Personal computer
Plotters
Printers
Programming
Spreadsheets
Users
Word processing