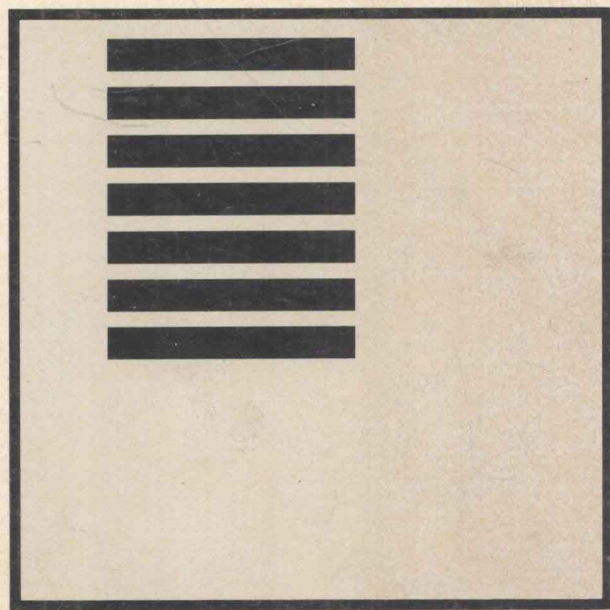


System Design Guide

featuring dBASE II®

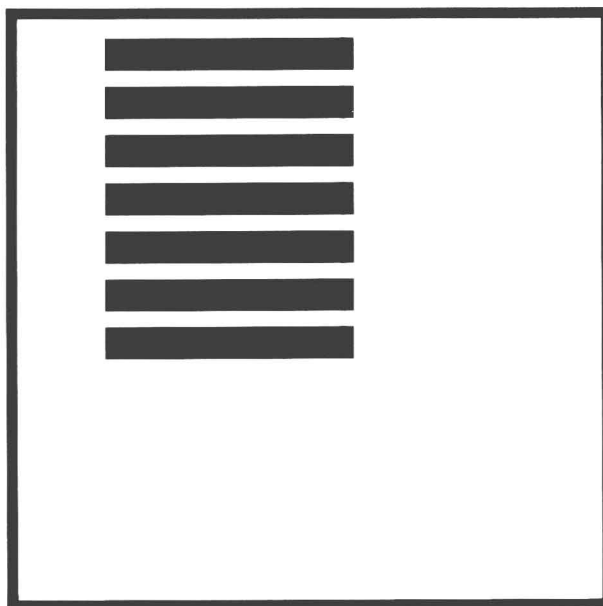
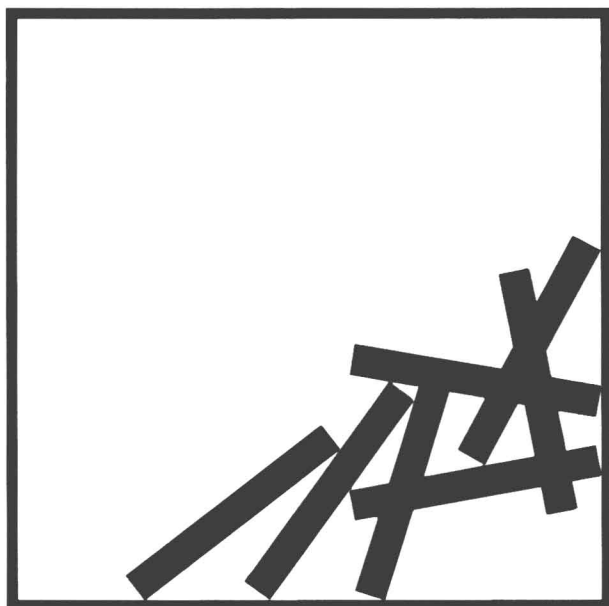
by Ron Freshman



System Design Guide

featuring dBASE II®

by Ron Freshman



Managing Editor: Sandra Rochowansky
Editor: Robert Hoffman
Production Manager/Text Designer: Thomas Clark
Cover Design: Robert D. Ross
Illustrations: Vadim Gorenbeyn

ASHTON · TATE ■

10150 W. Jefferson Boulevard
Culver City, CA 90230
(213) 204-5570

dBASE II is a registered trademark of Ashton-Tate
IBM is a registered trademark of the
International Business Machines Corporation
CP/M is a registered trademark of Digital Research, Inc.

Copyright © 1984 Ashton-Tate. All rights reserved.
ISBN: 0-912677-12-0



ACKNOWLEDGEMENT

I would like to thank all those who have helped bring this book to fruition. To Jane Mellin and Robert Hoffman for the gentle manner with which they applied their editing talents. To Bill Jordan for the encouragement and support that he provided. The insightfulness of D.A. Gray has provided me with a much simpler way to view the organization of technical material. To the extent that she was successful in making me understand the beauty in simplicity, we can all be thankful. I am especially grateful to my wife, Adrienne, for her support and frank yet subtle suggestions along the way.



PREFACE

The *System Design Guide* is an introduction for people who would like to develop a computerized system to solve a paperwork information problem. The *Guide* discusses the general approach to problem analysis and the specific steps required to design and implement a system. *dBASE II* is the tool used to develop a “starter-kit” system, which both solves a particular problem and can be used as the base from which to develop other applications. It is a basic system with a flexible structure that can be easily modified to meet other requirements.

For those readers already experienced in system design, a *Fast-Path* is included. The highlighted portions of the text provide a quick introduction to *dBASE II*.



TABLE OF CONTENTS

1

CHAPTER ONE

Introduction

10

CHAPTER TWO

Basic Terminology

20

CHAPTER THREE

Select and Define the Problem

38

CHAPTER FOUR

System Design

77

CHAPTER FIVE

Personal/Business Directory, Version I



TABLE OF CONTENTS

100

CHAPTER SIX

Personal/Business Directory, Version II

126

CHAPTER SEVEN

What to Do Next

132

APPENDIX ONE

Blank Documentation Forms

137

APPENDIX TWO

Screen Editing

142

APPENDIX THREE

System Documentation Manual



1

CHAPTER ONE INTRODUCTION

1.

INTRODUCTION

2.

WHO SHOULD READ THIS BOOK

3.

WHY *dBASE II*?

4.

HOW TO USE THIS GUIDE



2

CHAPTER ONE INTRODUCTION

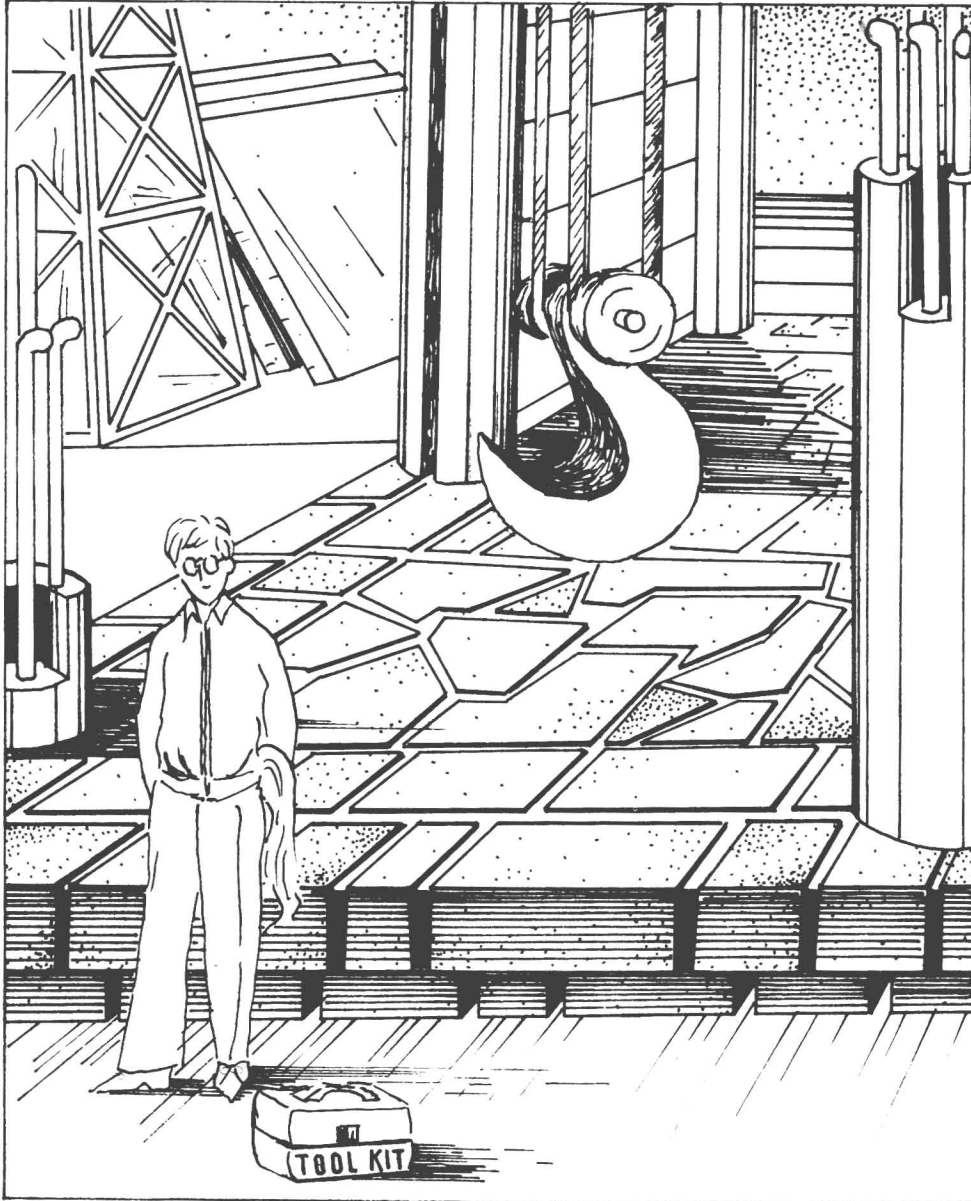
A database system is the computerized solution to a paperwork information problem. With the *System Design Guide* you will select and analyze a problem, and then develop its solution on several levels of sophistication. You'll learn how to approach any information problem, as well as how to turn a paperwork problem into a computerized solution. The *Guide* is an introduction for people who would like to develop systems but do not have the time or the inclination to become computer programmers.

Any project is simplified once the proper foundation is in place. Obtaining the knowledge to construct the foundation is also easy when the building blocks are laid one by one in a logical order. Unlike a computer manual, which provides a detailed description of a software program's capabilities, this guide is structured to give you only the information required to master each topic. Each section of the guide is like a building block, necessary to complete your foundation for building a computerized system.

The *System Design Guide* provides the building blocks and *dBASE II* provides the tools. Your toolkit—the commands necessary to implement a system—is demonstrated by stepping through the design and development of a “starter-kit” system. Later versions of the starter-kit are enhanced with other *dBASE II* capabilities, making it easier to use and insuring the correctness of the stored data.



3



**ANY PROJECT IS
SIMPLIFIED ONCE THE
FOUNDATION IS IN PLACE**

4

CHAPTER ONE INTRODUCTION

WHO SHOULD READ THIS BOOK

A programmer spends most of his or her time considering computer idiosyncrasies and learning the specific rules of programming languages. The *System Design Guide* will help you understand a problem and then design a computerized solution without getting mired in the programming. This book is for people who have paperwork and information problems, including:

- Individuals who want to computerize their paperwork problems without becoming programmers.
- People with the skill and experience to develop software applications but who want a concise introduction to designing *dBASE II* solutions.
- People who know that they are in the midst of an information explosion with paperwork and forms piling up faster than they can deal with them, but are not sure how a computer can help.

Let's examine these groups and see how the *Design Guide* can help.

Those of you who need to develop a specific computer application but do not want to become computer programmers are the primary audience for the *Guide*. You have three options when computerizing an application. You can purchase programming services, buy an applications software package, or use a tool such as *dBASE II* that facilitates communication of specific requirements directly to the computer.



5

CHAPTER ONE INTRODUCTION

If you hire a programmer to write a custom program, you could find yourself in a “Catch-22” situation. In order to program the computer according to your specific requirements, the programmer must understand your business. The catch is that no one can quite understand the information requirements of your business in the same way that you do. A programmer’s main task is mastering a never-ending stream of complex concepts, new system rules, and cryptic codes. Meeting the technical demands of data processing does not leave much time to understand the user’s real requirements. Even a programmer experienced in your field will have to spend a considerable amount of time (and your money) learning your specific requirements. Do not feel alone. Most large corporations have this problem. The question is, should business people be trained in programming or should programmers learn the business? Whether you develop your own system or hire a programmer, this *Guide* will help you communicate your needs and monitor their satisfaction.

Buying a complete applications package is an excellent choice, if the package can meet most of your requirements. Usually, the package is available for review. You also know, in advance, the exact cost of the software—something you can only estimate with other approaches. But the major problem with an applications package is that you are limited to what is offered. If you require a feature that is not provided, there is little chance of adding it to the package. Generally, you purchase programs that can only be executed on a computer (object code); modifiable program instructions (source code) are not usually available.



6

CHAPTER ONE INTRODUCTION

If you have a unique application, or you want to customize a system, you have the third alternative of using a software tool like *dBASE II*. This will let you communicate directly to the computer without going through a programmer. Here you specify the system features and select the instructions to the computer (commands), thus the limitations of an applications package are removed. This is not, however, a “free lunch.” There’s still much work necessary on your part, but at least you can avoid unnecessary technical complexities.

WHY *dBASE II*?

Prior to Database Systems like *dBASE II*, computer languages such as BASIC, COBOL, Pascal, C, and Assembly languages were the only means available to develop computer systems. You certainly would be considered a programmer if you developed computer systems with one of these languages. They require attention to lots of technical details. Database development systems, on the other hand, provide the necessary tools for straightforward system development. They remove many of the technical complexities and let non-programmers develop their own projects.

dBASE II was selected from the several available Database Systems because of its flexibility and completeness. It has an extensive list of powerful, English-like commands that are easy to use. You can store and manage data, prepare reports, and handle screens. Additionally, any files created with *dBASE II* are easily modifiable—to meet your changing information needs. Virtually



7

CHAPTER ONE INTRODUCTION

any paperwork problem that fits within the size and time constraints of a personal computer can be answered using *dBASE II*.

For programmers, the great flexibility of *dBASE II* is clearly an advantage. To develop a system with conventional programming languages, the largest share of time is devoted to writing programs just to manage the data: storing, querying, displaying, and reporting. With *dBASE II*, they are freed from many of these tedious and time-consuming chores. Programmers can be more responsive to their client's unique requirements for a much more reasonable cost—a blessing for both programmers and users alike.

On the other hand, *dBASE II*'s wide range of capabilities can sometimes be a stumbling block to non-programmers: there is so much to learn. The manuals, texts, and training classes are extensive but do not provide a direct path to problem-solving for many individuals. For these people, the *System Design Guide* bridges this information gap and permits non-programmers to effectively use a comprehensive database system. The *Guide* contains step-by-step instructions and procedures to organize and structure paperwork problems so that they provide a foundation for effective computer systems.

HOW TO USE THIS GUIDE

You must know how to turn on the computer and you must have *dBASE II* configured to your computer. If you haven't performed these tasks, consult your computer manual and the *dBASE II User's Guide*. That is, you must be at the point where you



8

CHAPTER ONE INTRODUCTION

can get the “dot prompt” on your computer screen. It will be easier if you have had experience with the computer keyboard and the control key capabilities. The use of a word processor or editing software program will have provided this experience. The *System Design Guide* will take you from this point on to the building of computerized application systems.

Although you will learn enough *dBASE II* commands to develop a system, not all of the *dBASE II* capabilities are examined. The *Guide* is not intended to be a tutorial on *dBASE II*. It will be necessary to refer to the manual supplied with *dBASE II* for a comprehensive study of the software’s range.

Non-programmers can proceed through each chapter, step-by-step, to define the problem, design the solution, and then enter each command required to develop the starter-kit system.

A *Fast-Path* method is provided for programmers with experience developing systems, who would like to see how a system can be quickly developed using *dBASE II*. They can scan the text and enter the highlighted *dBASE II* commands to develop the starter-kit system. This provides a quick guide to the essential *dBASE II* commands and establishes a structure upon which the more advanced commands can be effectively used.

Those who are unsure how a computer can help can read through the sections on Defining the Problem and System Design to see if his or her particular problem fits within available resources. If, at that point, it appears that a computer system might provide some help, but you are unwilling or unable to develop an



9

CHAPTER ONE INTRODUCTION

application, you might pass the work already done on the starter-kit system and the documentation aids to the system developers. These items could reduce your implementation costs. The definitions of database terms and concepts will also be of help in working with system developers and programmers.

In addition, users of corporate data processing departments for their information systems can use a microcomputer and *dBASE II* for “prototyping.” You can build a model of what you would like from the corporation’s mainframe computer. This model will provide more explicit communication of your needs to your data processing group and can speed up implementation of the major system.



10

CHAPTER TWO BASIC TERMINOLOGY

1.

DATABASE SYSTEM

Database File

Field

Record

Key Field: Related Databases

2.

SYSTEM COMMANDS

Note About Filetypes

3.

TECHNICAL TERMS

