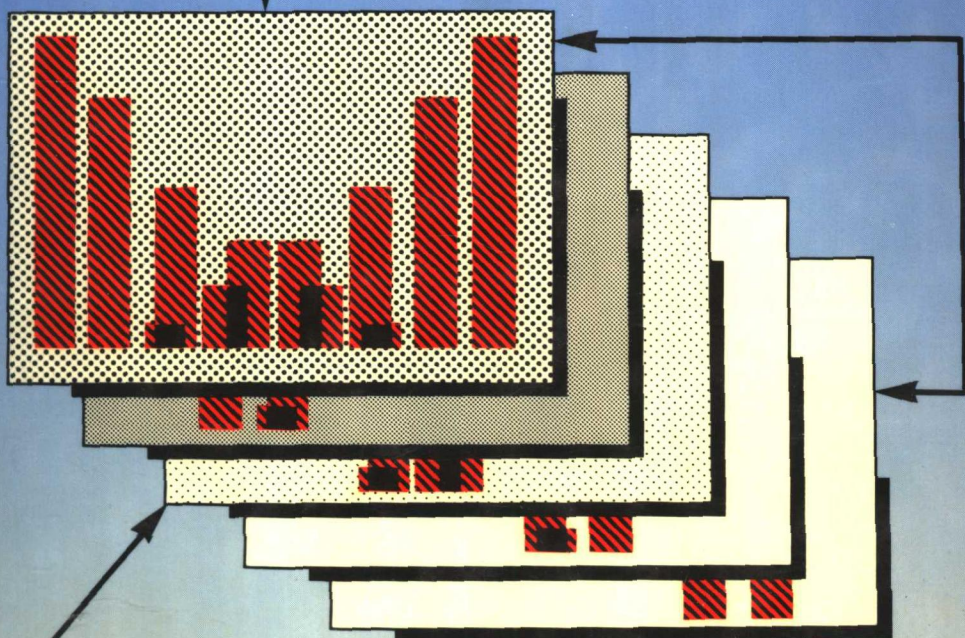


PRACTICAL BUSINESS SYSTEMS IMPLEMENTATION

A DP GUIDE

ALEX VARSEGI



PRACTICAL BUSINESS SYSTEMS IMPLEMENTATION A DP Guide

Alex Varsegi



VAN NOSTRAND REINHOLD COMPANY

New York

Copyright © 1988 by Van Nostrand Reinhold Company Inc.
Library of Congress Catalog Card Number 88-5597
ISBN 0-442-28843-3

All rights reserved. No part of this work covered by the copyright hereon may be reproduced or used in any form or by any means—graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems—without written permission of the publisher.

Printed in the United States of America

Van Nostrand Reinhold Company Inc.
115 Fifth Avenue
New York New York 10003

Van Nostrand Reinhold Company Limited
Molly Millars Lane
Wokingham, Berkshire RG11 2PY, England

Van Nostrand Reinhold
480 La Trobe Street
Melbourne, Victoria 3000, Australia

Macmillan of Canada
Division of Canada Publishing Corporation
164 Commander Boulevard
Agincourt, Ontario M1S 3C7, Canada

16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1

Library of Congress Cataloging-in-Publication Data

Varsegi, Alex.
Practical business systems implementation.

Includes index.

1. Business—Data processing. 2. System design.

I. Title.

HF5548.2.V355 1988 658'.05 88-5597

ISBN 0-442-28843-3

*To my wife, Marie
To my children, George and Mary
and to the memory of Brownly Walter*

PREFACE

Whether you need to design your own system or simply install a vendor product—this is the book you need. It will give you an understanding of the real systems issues involved, and a clear and concise presentation of all the up-to-the-minute productivity tools.

Practical Business Systems Implementation is written for professionals with previous exposure to data processing concepts in a main frame computer environment. It is a valuable tool for both programmers and systems analysts, and a must for project leaders/managers who have responsibility for implementing a major system. It will also be useful to students and recent graduates of computer science. Today's computer science graduates have a basic understanding of systems design and systems analysis, but did not learn how to apply vendor packages. Vendor package implementation, which represents over 70 of all new business systems, is critical for all computer science students.

Finally, the book will be useful also to the executive with some technical background who wants to learn about some of the state-of-the-art productivity tools available on the market. *Practical Business Systems Implementation* is a unique approach to systems design and analysis, as well as to Project Management. It deals mainly with on-line real-time topics, although some discussion of batch consideration is included. Many vendors have been interviewed and the book contains a wealth of proprietary vendor material that has never been published before. Finally, the book is distinguished from many other books on how to design a system, by its emphasis on how to get by without a "start from scratch" design and achieve results many times more cost effective through the use of vendor packages.

I would like to express my appreciation to a number of companies without whose help this book would not have been completed. Among these are Applied Data Research, Index Technology, The Pansophic Company, Compuware, On-Line Software, Inc., the McCormack & Dodge Corporation and Software A.G. Inc.

Figure 1 illustrates the organization of the book. Chapter 1 is a brief

overview of the Requirements Study and a brief evaluation of some of the prototyping technologies available. Among these are IBM's "C-LIST" or Dialog technique, and Index Technology's "Excelerator" products. I recommend that you read this chapter carefully regardless of your approach to systems analysis.

If you are in the process of evaluating the advantages and the drawbacks of an in-house system you should read Chapter 2. If a decision to acquire a vendor product is imminent, you might prefer to skip Chapter 2 and concentrate exclusively on Chapters 3, 4, and 5.

Whether your need is to make a decision on in-house versus a vendor product, you want to concentrate on Chapter 6 which is an overview of Project Status Reporting Methodologies. Chapters 7 through 10 focus on the required activities in an in-house system. The approach taken here is on how you can accomplish success in this technical area. Finally, Chapters 11 through 14 give you a consolidated overview of tasks that are common to both in-house and a vendor package.

Throughout these chapters, I've heavily concentrated on top quality products which I am comfortable in recommending to you.

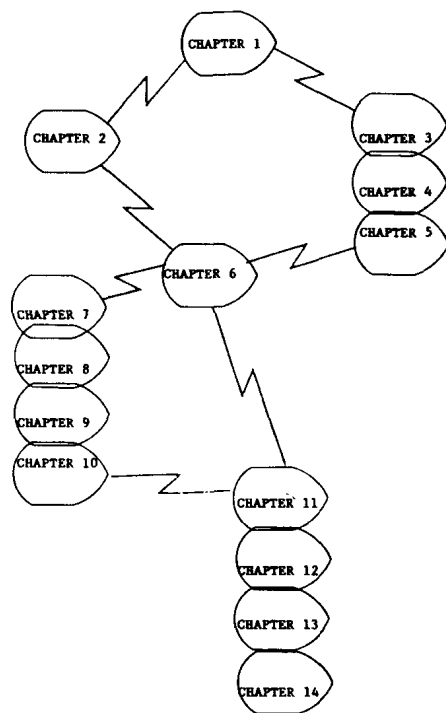


Fig. 1. A logical overview of the chapters

CONTENTS

PREFACE	/ vi
1. THE REQUIREMENTS STUDY	/ 1
1.1 Overview	/ 1
1.2 Internal or Mandatory Systems Requirements	/ 1
1.3 A Reiterative Process	/ 3
1.3.1 Scope and Conceptualization	/ 4
1.3.2 Constraints, Idiosyncrasies, Systems Questionnaire	/ 4
1.3.3 Output Definition—Prototyping the Easy Way	/ 5
1.3.4 Early Stages of Data Requirements	/ 12
1.3.5 Preliminary Thoughts on Customizing	/ 13
1.4 Prototyping Methods: Main Frame vs Microcomputers	/ 14
1.5 Inhouse Versus Vendor Approach	/ 16
1.6 Summary	/ 20
1.7 Review Questions	/ 22
2. WHAT TO LOOK FOR IN AN INHOUSE SYSTEM	/ 23
2.1 Overview	/ 23
2.2 Freezing the Specifications	/ 24
2.3 Know Your Deliverables	/ 25
2.4 Checkpoint Reviews	/ 27
2.5 More on Prototyping	/ 28
2.6 Changes and Standards	/ 31
2.7 Playback Module (Compuware)	/ 32
2.7.1 Data Capture	/ 34
2.7.2 Script Modifications	/ 35
2.7.3 Script Execution	/ 35
2.7.4 Output Comparison	/ 30
2.8 Summary	/ 37
2.9 Review Questions	/ 37

x CONTENTS

3. WHAT TO LOOK FOR IN A VENDOR PACKAGE—PART I	/ 39
3.1 Overview	/ 39
3.2 Structural Requirements	/ 40
3.2.1 The Exit Point Concept	/ 40
3.2.2 Shadow Files	/ 42
3.2.3 Screen Generators	/ 45
3.2.4 Natural, An Efficient Screen Generator Alternative	/ 51
3.2.5 Help Screens	/ 53
3.2.6 Menu-Driven Systems	/ 57
3.2.7 Modular or Borderless Design	/ 61
3.2.8 On-Line Security and Limited Access	/ 63
3.3 Summary	/ 65
3.4 Review Questions	/ 66
 4. WHAT TO LOOK FOR IN A VENDOR PACKAGE—PART II	 / 68
4.1 Overview	/ 68
4.2 The Need to Customize and How Much	/ 69
4.3 Self-Diagnosing Systems	/ 73
4.4 Table-Driven Systems	/ 74
4.5 Vendor-Initiated Implementation Schedule	/ 76
4.6 Summary	/ 79
4.7 Review Questions	/ 80
 5. STATE OF THE ART IN VENDOR TECHNOLOGY	 / 81
5.1 Overview	/ 81
5.2 The Millennium Product—McCormack & Dodge	/ 82
5.3 Data Organization	/ 83
5.4 The Command Line Concept	/ 85
5.5 Millennium-Based Help Functions	/ 91
5.6 Link Levels and the Sticky Cursor	/ 94
5.7 Maintaining and Querying Data	/ 95
5.8 Summary	/ 99
5.9 Review Questions	/ 101
 6. PROJECT STATUS REPORTING	 / 102
6.1 Overview	/ 102
6.2 Planning and Controlling a Project	/ 102
6.3 Using the Gantt Chart	/ 105
6.4 The Profile of a Project Manager	/ 107
6.5 Critical Path Method (CPM)	/ 108
6.6 Program Evaluation Review Techniques (PERT)	/ 109

6.7	Reporting on the Project Status—Micro Versus Mainframe Packages	/ 112
6.8	Summary	/ 113
6.9	Review Questions	/ 116
7.	GENERAL OR CONCEPTUAL DESIGN	/ 117
7.1	Overview	/ 117
7.2	The Hierarchical Structure	/ 119
7.3	IMS, a Product of IBM	/ 124
7.4	The Network Structure	/ 131
7.5	The Relational Structure	/ 132
7.6	DATACOM, a Product of ADR	/ 140
7.7	IBM—DB2 Overview	/ 148
7.8	The Functional Model	/ 150
7.9	Summary	/ 152
7.10	Review Questions	/ 154
8.	DETAIL DESIGN—PART I	/ 155
8.1	Overview	/ 155
8.2	Developing Input/Output Panels	/ 155
8.3	IBM Screen Development Facility (SDF)	/ 163
8.4	File Design, Data Dictionary	/ 174
8.5	Summary	/ 177
8.6	Review Questions	/ 178
9.	DETAIL DESIGN—PART II	/ 179
9.1	About Telecommunication Monitors	/ 179
9.2	The On-Line Processing Cycle	/ 182
9.2.1	The Add Mechanism	/ 183
9.2.2	The Delete Mechanism	/ 186
9.2.3	The Revise of Update Mechanism	/ 187
9.2.4	The Inquiry Mechanism	/ 189
9.3	Writing Program Specifications	/ 189
9.4	Summary	/ 194
9.5	Review Questions	/ 200
10.	ON-LINE PROGRAMMING METHODS	/ 201
10.1	Overview	/ 201
10.2	How to Establish Access to the Keyboard	/ 202
10.3	How to Handle “Triggered” Conditions	/ 204
10.4	How to Establish On-Line Storage Areas	/ 206

xii CONTENTS

10.5 Linking, Exit Controlling, and Returning /	211
10.6 Efficiency in Writing On-Line Code /	213
10.7 Summary /	215
10.8 Review Questions /	215
 11. INTERFACE /	217
11.1 Overview /	217
11.2 Developing a Batch Interface /	218
11.3 Developing an On-line, Realtime Interface /	219
11.4 Security for On-line Interfaces /	224
11.5 Case Study /	226
11.6 Summary /	231
11.7 Review Questions /	232
 12. TEST ENVIRONMENTS AND PRODUCTIVITY TOOLS—PART I /	233
12.1 Overview /	233
12.2 Test Cycles /	234
12.2.1 Unit Test /	234
12.2.1.1 Testing in an Incomplete Environment /	235
12.2.1.2 Unit Testing in a Complete Environment /	237
12.2.2 Integration Test /	239
12.3 Acceptance Test /	245
12.4 Panvalet Library Procedures /	245
12.5 Documentation /	250
12.5.1 Systems Documentation /	250
12.5.2 User Documentation /	251
12.5.3 Operational Documentation /	252
12.6 Summary /	253
12.7 Review Questions /	253
 13. TEST ENVIRONMENTS AND PRODUCTIVITY TOOLS—PART II /	255
13.1 Overview /	255
13.2 Case Study 1: EDF Versus INTERTEST /	257
13.3 Case Study 2: ABEND-AID /	265
13.4 Summary /	270
13.5 Review Questions /	271
 14. THE CONVERSION PROCESS /	272
14.1 File Conversion Review /	272
14.2 Converting to a New Access Method (ISAM to VSAM) /	284
14.3 Converting to an Advanced Language (ADR) /	288
14.4 Migration Software (ADR) /	294

14.5 Paralleling /	298
14.6 Summary /	299
14.7 Review Questions /	299

GLOSSARY /	301
------------	-----

INDEX /	307
---------	-----

1

THE REQUIREMENTS STUDY

Overview
Internal or Mandatory Systems Requirements
A Reiterative Process
Scope and Conceptualization
Constraints, Idiosyncrasies, and the Systems
Questionnaire
Output Definition—Prototyping the Easy Way
TSO-ISPF Dialog Method (IBM)
Early Stages of Data Requirements
Preliminary Thoughts on Customizing
Prototyping Methods: Mainframes versus Micro-Computers
Inhouse versus Vendor Approach
Summary
Review Questions

1.1 OVERVIEW

The purpose of this chapter is to provide an overview of a REQUIREMENTS STUDY. Also presented is a series of related topics to highlight the ever-increasing value of prototyping.

Prototyping is a method of emulating a proposed system without costly programming or design efforts. In a way, it is analogous to modeling techniques. Prototyping allows you to determine the consequences of certain actions. Thus, using prototyping, you can project the workings of a system and determine its success or shortcomings in an embryonic stage of development.

Prototyping is rapidly becoming an important tool in all cycles of a system but is especially valuable in doing the Requirements Study.

1.2 INTERNAL OR MANDATORY SYSTEMS REQUIREMENTS

Let us consider the importance of your current computer environment. Vendor packages are normally restricted in the sense that they are geared to a

2 PRACTICAL BUSINESS SYSTEMS IMPLEMENTATION

specific computer environment, or perhaps a few selected environments, in terms of both hardware and software assistance.

Suppose your current environment encompasses an IBM 4381 mainframe under OS/MVS. The system utilizes IDMS/R database technology monitored via Customer Information Control System (CICS). The application language used throughout the department is ANS COBOL, although for some of the more simple applications, EASYTRIEVE and MARK IV are also utilized. The access method was recently converted from the Indexed Sequential (ISAM) to the Virtual Storage Access Method (VSAM), with most of the source codes modified to accommodate the new requirements. Finally, most users have access to an IBM 3178-type CRT for data entry and inquiries.

Against this background are three vendor packages. Their features are summarized in Fig. 1.1. All three packages run on the IBM 4381 mainframe. However, the third package cannot function in the current computer environment because of an incompatible operating system, a Disk Operating System (DOS).

From a practical point of view, you probably don't want to run or even emulate two vastly different operating systems on your computer, which otherwise could be accomplished using the Virtual Machine (VM) concept. Thus, vendor 3 is disqualified immediately. Let's assume now that Vendor 1 in Fig. 1.1 does not accommodate the IDMS/R database structure. In fact, this vendor doesn't even require a formal database environment in which to operate. In addition, Vendor 1's system is a non-CICS system, i.e.,

	VENDOR 1	VENDOR 2	VENDOR 3
Does it run on an IBM 4381?	yes	yes	yes
OS/MVS compatible?	yes	yes	no
IDMS/DATABASE driven?	no	yes	no
CICS telecommunications monitor?	no	yes	no
Application language: COBOL?	yes	yes	yes
Can EASYTRIEVE access files?	yes	yes	yes
Can MARK IV access files?	yes	yes	yes
Is access method VSAM compatible?	yes	yes	yes
Can IBM 3178 C-type CRTs be used?	yes	yes	yes

Fig. 1.1 A set of internal requirements (far from being complete . . .) matched against three vendor packages.

a package utilizing its own monitor. (Actually, many vendors once found it convenient to develop their own telecommunications monitor in order to be more generic in accommodating clients with different computer environments.)

As you can see, by simply reviewing some of the initial premises of your operating environment, you can immediately eliminate vendor products that are totally or fairly incompatible with your physical requirements. In the above situation, two out of three packages were eliminated without the need to involve a large number of decision makers.

Whether the source code of the remaining vendor package is ANS COBOL, PL/I, or another language is unimportant, since all the customization must be achieved in vendor-established areas referred to as EXIT POINTS. (This topic will be discussed further in Chapter 3). So, instead of being concerned about the programming language used by the vendor, you should focus on the availability of Exit Points throughout his system.

1.3 A REITERATIVE PROCESS

Let me now begin with what I perceive to be the logical sequences of a Requirements Study. The first step, which is normally a monumental task, is to interview key users in order to determine their requirements. How important are these sessions? What should be one's attitude throughout the process?

The interview process is crucial, since you are in fact developing a working relationship with the various users. The manner in which you conduct yourself will eventually influence the attitude and involvement of users throughout the entire course of the implementation process. This highlights the importance of the first impression, which is the most critical phase in establishing a cooperative environment.

The following procedures are useful in conducting interviews:

- Always contact the highest-ranking member of a given unit or department, even if he or she is not the individual you need to interview.
- Broaden the scope of your interviews to include an optimal number of participants, even personnel who are no longer connected with the project.
- Do your homework before each interview session. Always prepare an agenda, and gear it to the user's level of expertise and understanding.
- Get into the habit of preparing minutes after each interview session to verify that what was discussed during the meeting was in fact a mutual accomplishment, not merely your own interpretation of it. If, for some reason, you have completely missed or misunderstood a topic, you should

4 PRACTICAL BUSINESS SYSTEMS IMPLEMENTATION

probably reissue your memo to reflect the change. It's good practice (as well as courteous) not to publish anything until it has been proofread by the participating user.

But what about some of the obvious inconsistencies among users representing a variety of interests? Suppose one user wholeheartedly agrees with (even endorses) a particular set of procedures, or approach, while others are vehemently against it? You're certainly welcome to address these problems, but it is not your responsibility to resolve them. The best procedure is to secure a sign-off following each interview session, based on your minutes, whether there is general agreement or not.

Let us now continue the discussion of the general topic, the Requirements Study, using a reiterative approach. This method will be used to further refine every task that needs to be addressed in this process.

1.3.1 Scope and Conceptualization

Let's outline some of the basic requirements, for example, in developing a Payroll Module to handle both delayed and anticipated employees. The general requirements should, of course, include a biweekly and a semi-monthly pay cycle, as well as a mechanism to handle the different tax computations for a number of states. Also, the system must rely on remote telecommunications to link up and interface with various local environments. To further expand on this process, we will begin with a conceptual overview, which is presented in Fig. 1.2. This is the first step, or initialization cycle, in the overall process.

1.3.2 Constraints, Idiosyncracies, and the Systems Questionnaire

During the follow-up step or second cycle, the primary objective is to resolve a number of problems concerning the constraints and idiosyncrasies of the system. This is helpful in reviewing the existing procedures in order to transform a general approach into a more detail-oriented process. To accomplish this, I prepared a sample questionnaire that will help to reveal the pluses and the minuses of the inhouse design compared to those of various vendor alternatives. A comprehensive sample of this is shown in Fig. 1.3 The questionnaire is also valuable in further evaluating the contents or prerequisites of competing vendor packages.

For an inhouse system, since there is no such system at present, there will be little or no problem in addressing (or accommodating) a majority of the requirements—provided the budget is sufficient. The vendor's software, of course, is another story. It is a finished product, packaged and wrapped,

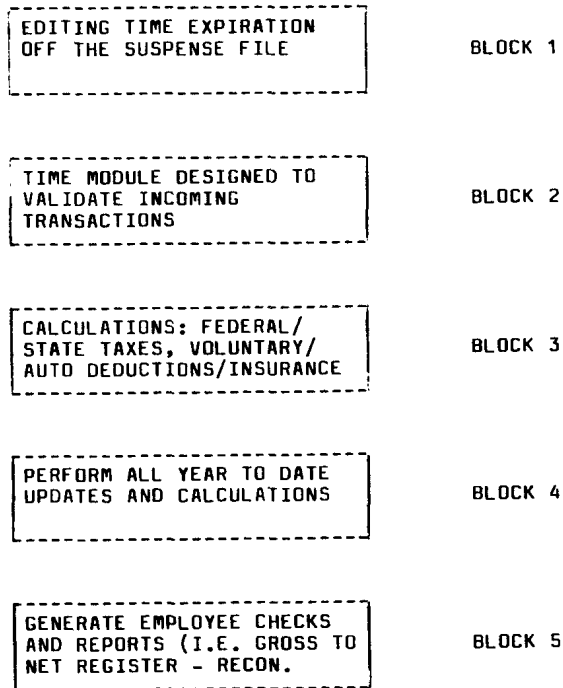


Fig. 1.2 A conceptual review of certain main events.

and any new requirements outside of its parameters could be a potential problem. In reality, documentation of this sort may be thirty to forty pages in length. The additional column on the questionnaire contains the vendor's response, where such a response is in order.

Such a questionnaire may contain a number of areas. The payroll system portion of it is only one of the many formats used in the evaluation. Additional areas may include interface requirements, systems and general operating topics, etc. These topics may be further subdivided, depending on the scope of the project.

1.3.3 Output Definition—Prototyping the Easy Way

At this stage, you are ready to enhance your skills in summing up your output requirements by introducing some prototyping features. The more skillful you are in applying the various prototyping methods in your work,

6 PRACTICAL BUSINESS SYSTEMS IMPLEMENTATION

SPECIFICATIONS QUESTIONNAIRE	RESPONSE	REFERENCES
3.1.2 On-Line Processing		
On-Line Implementation Support		
Training Documents		
Are training documents provided for our application programmers?	YES	SYSTEMS DOC
Are training documents provided for our technical group?	YES	SYSTEMS DOC
Are training documents provided for our user?	YES	SYSTEMS DOC
3.1.2.1 Classroom Training		
a. Is classroom training provided for our application programmers:		
At our location?	YES	PER CONTRACT
Within the Chicago area?	YES	PER CONTRACT
b. Is classroom training provided for our technical group:		
At our location?	YES	PER CONTRACT
Within the Chicago area?	YES	PER CONTRACT
c. Is classroom training provided for our user:		
At our location?	YES	PER CONTRACT
Within the Chicago area?	YES	PER CONTRACT
d. Is classroom training provided for new releases to your system:		
At our location?	YES	PER CONTRACT
Within the Chicago area?	YES	PER CONTRACT
e. Do you provide an implementation team to help		
Prepare files, programs, and operating procedures?	YES	PER CONTRACT
Prepare systems documentation?	YES	PER CONTRACT
f. Do you have a standard conversion plan that can be modified for this installation?	YES	
g. Do you provide personnel on site during the first weeks following conversion to answer questions and to ensure smooth operations of the system?	YES	PER CONTRACT

Fig. 1.3 Part of a Specifications Questionnaire.