
A
BOOK

AN INFORMATION SYSTEMS MANIFESTO

Library of Congress Cataloging in Publication Data

Martin, James
An information systems manifesto.

Includes bibliographical references and index.
1. Electronic data processing. 2. Management information systems. I. Title.
QA76.M327 1984 658.4'0388 83-17730
ISBN 0-13-464769-6

An Information Systems Manifesto
James Martin

Editorial/production supervision by Linda Mihatov
Front matter design by Judith A. Matz
Jacket design by Diane Saxe
Manufacturing buyer: Gordon Osbourne

Printed in the United States of America

10 9 8 7 6 5 4

ISBN 0-13-464769-6

PRENTICE-HALL INTERNATIONAL, INC., *London*
PRENTICE-HALL OF AUSTRALIA PTY. LIMITED, *Sydney*
EDITORA PRENTICE-HALL DO BRASIL, LTDA., *Rio de Janeiro*
PRENTICE-HALL CANADA INC., *Toronto*
PRENTICE-HALL OF INDIA PRIVATE LIMITED, *New Delhi*
PRENTICE-HALL OF JAPAN, INC., *Tokyo*
PRENTICE-HALL OF SOUTHEAST ASIA PTE. LTD., *Singapore*
WHITEHALL BOOKS LIMITED, *Wellington, New Zealand*

PREFACE

An Information Systems Manifesto is designed to provide end users, DP staff, and senior and top management with a strategy and direction on how to change and manage the dramatically changing environment of information systems and data processing.

The revolution described in information systems is already in progress. Yet many corporations ignore it or fail to recognize the impact it will have on their organizations. The intent of this book is to provide people with the information they need to direct and implement these changes successfully.

The first part of the book describes the actions that should be taken to improve the use of computers, for example, by placing the end user firmly in the driver's seat and automating the process of automation.

The second part of the book examines the future, particularly the Japanese and the essential need for entrepreneurs.

This book was designed for a wide cross-section of people and is uncluttered with technical details.

It should be mandatory reading for *everyone* concerned with information systems and data processing.

James Martin

ACKNOWLEDGMENTS

The author wishes to thank the consultants of James Martin Associates for their contributions to this book and their ongoing work in implementing the methodologies it describes.

CONTENTS

Table of boxes

PART I THE PRESENT

1 THE CRISIS IN DATA PROCESSING

PROBLEMS WITH DATA PROCESSING

The power of today's computing technology is not being used as it should be in most enterprises. Data processing is bogged down in problems. Managers are not receiving the information they require from their systems. Many decisions that should be made with the aid of computers are in fact being made with hand methods or inadequate information. Systems are so difficult to change that they often inhibit the implementation of new and important procedures that managers require. Computer users are increasingly hostile to DP but feel powerless to prevent the problems.

Among top-level managers there is often a sense of anger that they are spending so much on computing and yet seem unable to change procedures or obtain the information they need. In one corporation with an expensive and elegant worldwide computer network the chief executive complained bitterly to the author that for years he had been asking for daily or even weekly figures and cash balances but seemed no closer to obtaining these or other information he needed.

Box 1.1 lists the problems of computing as seen by users and management.

Failure to use computers effectively is serious on a national scale because it affects most aspects of productivity. This affects inflation and the ability to compete overseas.

The computer is the most flexible machine invented, capable of a staggering diversity of applications. It is rapidly dropping in cost and its power needs to be used as fully as possible for improving the efficiency of organizations. The problem lies not in the machine itself, but in the methods we use for creating applications. The traditional "application development life cycle" is slow and rigid. Its methods have been cast into concrete in many organizations with standards and procedures. But in many ways the procedures are not working.

BOX 1.1 Problems of computing as seen by users and management

- Users cannot obtain applications when they want them. There is often a delay of years.
- It is difficult or impossible to obtain changes that managers need in a reasonable amount of time.
- The programs have errors in them or sometimes do not work.
- Systems delivered often do not match the true user requirements.
- It is difficult to understand DP and communicate precise requirements.
- Specifications, on which users have to sign off, are difficult to check and are usually full of inconsistencies, omissions, and errors.
- Systems cost much more to develop and to maintain than anticipated.
- Because of the long time required to obtain results, the most important decision-support systems are never implemented. The support is needed more quickly than the time needed to create the programs.

This inability to use computers effectively should be regarded as a major organizational problem to which solutions *must* be found. The problem has reached crisis proportions. The solutions now exist but many organizations are not using them except in isolated pockets. They require new software, new procedures, and new management structures.

As we look at the history of technology we can observe certain times when a major break with the past methods had to occur. In computing, a set of application development methods has been accepted and slowly refined for more than two decades. We have now reached a point when these are inhibiting the most effective uses of computers. Fundamentally different methods are needed and are coming into use. Unfortunately, many data-processing (DP) organizations are not adopting the new methods rapidly enough.

Steam engines made earlier this century were beautifully intricate machines. They had many polished brass sliding rods, levers, and cams. Engineers had invented elaborate mechanisms for extracting a fraction of a percent of extra performance. They lovingly tuned the mechanisms and held technical symposia on steam engines long after the electric motor was in use.

Comparing relational data-base facilities with some of the old data-base management systems, or comparing new-architecture small computers with the old large operating systems, seems like comparing an electric motor with steam engines. The architects and maintainers of the old software are making

elaborate additions which give only minor improvements, when a switch to fundamentally better mechanisms is needed.

The difference between COBOL development and the creation of applications with the new application generators is even more dramatic. Many powerful new techniques exist today for creating software and building applications, but a million programmers and analysts around the world have not heard about them. Amazingly, in Europe steam engine designers were still refining these machines 50 years after the electric motor had been invented.

In Victorian factories with steam engines there were overhead shafts 100 feet long with large pulleys and belts going down to each machine tool. With electric power each machine tool could have its own motor. But the shafts remained in many factories long after their usefulness had ended. New tools often need fundamentally new methodologies. The procedures manuals in DP are often referred to as the "Bible" of DP development and it is heresy to disobey the Bible even when new software and techniques render it hopelessly obsolete. The old DP procedures, like the steam-driven shaft and pulleys, can preclude the freedom that is necessary to move flexibly with the new methods.

The *Random House Dictionary* defines a *manifesto* as "a public declaration of intentions, objectives, opinions or motives." In organizations everywhere a manifesto is needed from top management giving their policy for information systems. This can be translated into manifestos concerning the fundamental changes in DP techniques and productivity and how they should be handled by DP executives, systems analysts, and programmers.

APPLICATION BACKLOG

In most well-managed corporations the demand for new applications is rising faster than DP can supply them. The imbalance between demand and supply is steadily becoming worse.

Because of this, the backlog of needed applications is growing. Most corporations now have a backlog of from two to four years. One bank executive informed us that the bank's backlog was seven years. This situation is likely to become worse as machines drop in cost, unless better methods of creating applications are found.

The long backlog and inability of computer departments to respond to end users' needs quickly is very frustrating for the users. In many cases the users have felt that they cannot wait and have obtained their own departmental minicomputer or desktop computers.

INVISIBLE BACKLOG

Even though today's application backlogs are so long, they reveal only part of the story. When the documented backlog is several years (as it is in most installations), the end users do not even consider making requests for many of the applications they need. There is thus an *invisible backlog.*