The Personal Computer: Growth Patterns, Limits, and New Frontiers

Martin L. Ernst

Program on Information Resources Policy

Harvard University

Center for Information
Policy Research

Cambridge, Massachusetts

A publication of the Program on Information Resources Policy.

The Personal Computer: Growth Patterns, Limits, and New Frontiers

Martin L. Ernst
October 1991, P-91-6

Printing 5 4 3 2 1

October 1991

PROGRAM ON INFORMATION RESOURCES POLICY

Harvard University | **Center for Information Policy Research**

Affiliates

ABRH Consulting, Inc.
Action for Children's Television
American Newspaper Publishers Association
American Telephone & Telegraph Co.
Ameritech Publishing
Apple Computer, Inc.
Arthur D. Little, Inc.
Auerbach Publishers Inc.
Bell Atlantic
Bell Canada
BellSouth Corporation
Boice Dunham Group Inc.
Bull, S.A. (France)
Centel Corporation
CMC Limited (India)
Commission of the European Communities
Communications Workers of America
Computer & Communications Industry Assoc.
COMSAT
Cox Enterprises, Inc.
Department of Communications (Canada)
Dialog Information Services, Inc.
Digital Equipment Corp.
DRI/McGraw Hill
European Parliament (Luxembourg)
France Telecom
Gartner Group, Inc.
GTE Corporation
Hitachi Research Institute (Japan)
Honeywell, Inc.
IBM Corp.
IQ, Inc.
Information Gatekeepers, Inc.
Information Industry Association
International Data Corp.
International Monetary Fund
International Resource Development, Inc.
Invoco AB Gunnar Bergvall (Sweden)
I.T. Direction Ltd. (UK)
Japan Telecom Company
Kapor Family Foundation
Knowledge Industry Publications, Inc.
Korea Telecom
Lee Enterprises, Inc.
John and Mary R. Markle Foundation
Martin Marietta
McCaw Cellular Communications, Inc.
Mead Data Central
MITRE Corp.
National Telephone Cooperative Assoc.
The New York Times Co.
NEC Corp. (Japan)
Nippon Telegraph & Telephone Corp. (Japan)
Northern Telecom
Nova Systems Inc.
NYNEX
Ing. C. Olivetti & Co., S.p.A. (Italy)
OTC Limited (Australia)
Pacific Telesis Group
Public Agenda Foundation
Puerto Rico Telephone Co.
Research Institute of Telecommunications and Economics (Japan)
RESEAU (Italy)
Revista Nacional de Telematica (Brazil)
Salomon Brothers
Scaife Family Charitable Trusts
SEAT S.P.A. (Italy)
Siemens Corp.
Southern New England Telecommunications Corp.
State of California Public Utilities Commission
TEKNIBANK S.p.A. (Italy)
TFS, Inc.
The College Board
Third Class Mail Association
Times Mirror Co.
Tribune Company
United States Government:
- Department of Commerce
  - National Telecommunications and Information Administration
- Department of Defense
  - National Defense University
- Department of Health and Human Services
  - National Library of Medicine
- Department of State
  - Office of Communications
- Federal Communications Commission
- General Services Administration
- National Aeronautics and Space Administration
- National Security Agency
- U.S. General Accounting Office
- United States Postal Rate Commission

United Telecommunications, Inc.
US West
Williams Telecommunications
Wolters Kluwer

Acknowledgments

Thanks go to the following persons who reviewed and commented critically on drafts of this report: John A. Arcate, Herbert R. Brinberg, R.M. Gibson, John M. Hotchkiss, Mitch Kapor, Peter B. Maggs, Vincent Mosco, Mark S. Nadel, Jerry Wasserman, and Norman Weizer.

These reviewers and the Program's affiliates are not, however, responsible for or necessarily in agreement with the views expressed herein, nor should they be blamed for any errors of fact or interpretation.

Executive Summary

- A major factor in establishing the uses and values of PCs and workstations is the nature of the applications software available to operate them. In the first part of this paper, we examine the ways that PC and workstation software has evolved and improved in functionality over time, and the factors that establish limits on the pace of this evolution. We then consider the potential for new types of PC software that can be applied in areas of information processing where computers have had little use in the past. Finally, we suggest some likely impacts if such software begins to penetrate the market.

- It turns out that a specific pattern can be used to characterize the evolution of software, and to indicate in general terms what might be called the "natural paths" of development for an application. But carrying out the actions implicit in the growth pattern has depended on a parallel improvement in hardware capabilities. The improvement is needed in order to service simultaneously two general but conflicting needs: enhancement of the capabilities and scope of the individual application programs, and reduction in the complexity perceived by the programs' existing and potential users.

- The software creator's requirement to deal with the trend of increasing inherent complexity, as functionality is improved, is one of the factors that set limits on a given program's coverage and capabilities at any specific time. However, over the long term, the effects of these limits will tend to be quite weak. Helped by steady hardware advances, the ability to handle increased complexity also is growing, and programs for most major applications areas have natural growth paths that will continue to permit their expansion for the foreseeable future.

- While an understanding of how software packages evolve can give us a sense of their possible future growth paths, it does little to suggest new application areas. To cover this aspect of PC software growth, we turn to the activities that generate the needs computer programs seek to satisfy: the processes involved in using information. Here again, it is possible to develop a generic pattern of what goes on — we can define a set of processes and subprocesses that must be employed, in some order and often with iterations, whenever one seeks to benefit from information. From this we can show that

  - ▸ For a variety of institutional, technical, economic, and market-related reasons, PCs are used now to support only some of the different types of processing activities central to the use of information.

  - ▸ Quite specific possibilities exist for a family of related applications, based largely on searching and matching techniques, that have potential value in most of the processing activities where PCs have little or no usage.

- The new types of applications, if implemented, could alter the roles of PCs and workstations. They would become more continuously involved with their user's work, requiring less frequent but more skillful user attention, and performing a controlled portion of tasks that currently require human efforts.

- While there are many uncertainties to be faced, we already are in an excellent position to start testing the possibilities. These tests need not be expensive, and they could contribute immediate benefits to at least some users while providing guidance for future software design.

• The new types of software mentioned above can place new types of requirements on users. These requirements could lead to some long-term, and largely unanticipated, impacts on the skills that become most important for using PCs effectively. Our findings imply that

- There are a variety of limited, but quite significant, information processing activities for which computers can be "trained" (i.e., programmed) to replace human intervention in assessing the meaning and value of the substance contained in specific information items.

- To provide the needed training, the user must have (in addition to an appropriate level of knowledge of computers and programs) a deep understanding of language – of its capabilities, its limitations, and its local usage. In effect, the language requirement is not far different from the definition of what was needed to be fully literate in a classical sense.

If this situation arises, it will result from the growth of computer capabilities to where, for at least some (carefully monitored) purposes, they can manipulate words and phrases in a way that will give an outward appearance that they understand the symbolic content that the words and phrases are intended to convey. Given the fundamental level at which this type of change would be taking place, many other impacts are likely to follow.

Table of Contents

List of Figures

List of Tables

PART ONE

GROWTH PATTERNS AND LIMITS OF PC SOFTWARE

CHAPTER ONE

PERSONAL COMPUTERS AND THE "SOFTWARE BARRIER"

1.1 How We Got to Where We Are Now

For a number of years, *Datamation* magazine has conducted an annual survey of the mini- and microcomputer marketplaces. In its Executive Summary describing the 1989/1990 survey results, *Datamation* notes that four generic applications dominate the use of personal computers – word processing, spreadsheets, database management, and desktop publishing. Two of these applications (spreadsheets and word processing) were each being used on nearly 90 percent of all the PCs surveyed. The summary then goes on to mention that "no really great ideas" had been received among the responses to a question asking for possible new applications to supplant the two current leaders as "main engines of PC industry growth."

These comments should cause little surprise to most industry observers. The narrowness of the base of generic PC software applications with mass market potential has been evident since soon after these computers were introduced. From the start, the PC often has been a device looking for something worthwhile to do, particularly when being considered for uses in the home. Even at work, if you were a prospective "general purpose" user but had no strong reasons for personally typing documents or constructing spreadsheets, finding an acceptable formal justification for acquiring a PC could be very difficult.

If there is a surprise in this state of affairs at the turn of the 1990s, it may well derive from disappointment that optimistic hopes for dramatic new types of software have proven false in an industry that usually has thrived on being optimistic. Some added poignancy arises when this disappointment is contrasted with trends in performance and costs on the equipment side, where valuable improvements always seem to arrive right on schedule. As a result, and in spite of the dazzling advances in hardware, commentators have expressed concern that the personal computer industry is well on its way to becoming a mature, normal growth-rate industry. And we are beginning to hear again a plaint that the industry is "software-limited."

The role of software as a limiting influence has been with us since long before the introduction of the PC. It started in the very earliest days of computers – the days when programming had to be done in machine language, and each programmer had to write his own "sort" routines and other elementary functions. The response to challenges posed by these limits has led to much of the industry's progress: steady advances in the effectiveness of operating systems and the languages used for programming, the emergence of generic software packages and the growing use of software modules in applications development, and a host of other improvements in programming techniques and in the practices used to manage software development projects. And throughout these changes, vast increases in hardware capabilities – as well as parallel reductions in costs – were key to making progress. Up to the present, hardware limitations have been as critical as those imposed by software, although this usually is not obvious to the casual observer.

One result of the combined advances in hardware and software has been continuing change in both the operating environments of computers and the user populations that employ them. In the earliest period, manufacturers sold their computers "naked" – that is, without any associated software. The users were all professionals who wrote (or traded) their own programs (including most operating software). The market at this time was small, and concentrated in the fields of science and advanced engineering.

A second stage arrived with the earliest programming languages and operating software. Manufacturers now began to package their products, offering operating systems along with their hardware. Almost all aspects of hardware showed growth in performance, expanding the practical range of possible computer applications. As a result, significant government and commercial markets arose during these years, and this led to a shift in programming emphasis. Market economics had become more important, which made it critical to develop software that was highly efficient in its use of the scarcest physical resources; at the time, this normally was the high-speed memory. However, except for clerical workers preparing data inputs, most of the users still were professionals who had considerable knowledge of computers; and use of the early programs was confined to some form of batch entry operation.

A third period brought many advances, including CRT terminals, high capacity disk drives, and software that could support remote entry, time-sharing and real-time operations. These capabilities created a potential for major growth by broadening the markets for batch program applications and opening-up the markets for transaction processing, operations control, and database publishing systems. Programming became more difficult, and both operating systems and applications became far more complex. Response speed, reliability, and ease of maintenance started to compete with efficient use of resources for the top priority in software design. Not surprisingly, software costs began a long (and still continuing) climb as a fraction of total system costs. The higher costs, in turn, encouraged the success of the first generic software packages and the appearance of value-added resellers.

Another impact of the growth in software costs was that means to better organize and control the software development process became, and have remained, a matter of concern to senior management. At about the same time, the possibilities of using computers as an element of corporate competitive strategy began to attract attention at senior levels. Finally, the user population began to shift. Service agents and some corporate staff members, almost all of them lacking formal training or experience with computers, started interacting directly with the equipment. And managers at all levels became increasingly dependent on computer-generated reports.

The next phase brings us to the present. The introduction, and subsequent rapid improvement of PCs and workstations,[1] produced a period of massive growth in the number of computers in use. This growth, over time, led to a very large infusion of new, "hands-on" users and to the first mass markets for software products. However, even though they operated their own equipment, the large majority of the new users had little technical knowledge of computers and software and, often, not much interest in these subjects. Their concerns were focused almost entirely on the practical outputs that they sought. In response, software developers (and, to a lesser extent, hardware manufacturers) have been driven to emphasize certain features in their more recent offerings: ease of use, "friendly" user interfaces, and, sometimes with reluctance, a growing level of cross-application compatibility. In this way, the market is forcing considerable effective compatibility without resorting to the establishment of formal standards.

What this brief history indicates is that software always has been *one* limiting factor on computer industry growth; but it never was *the only* bottleneck. Opportunities to develop practical applications software have always been limited by the hardware capabilities available at a given time. Among the feasible possibilities, there have always been areas where economic trade-offs could be made between building more efficient software and using more powerful hardware. Similarly, there have always been potential trade-offs between building more "friendly" software and providing more user training. And, in each period, limitations were imposed by weaknesses in the software area whose alleviation depended on different forms of progress in other areas, as well as in software and software production techniques themselves. These features strongly suggest that any limitations for which

---

[1] There is no widely accepted and clear-cut definition of the differences between PCs and workstations. Some people have tried to draw a distinction by describing the PC as a piece of office equipment and the workstation as a device for professional workers, but this doesn't hold up well when examined in detail. Another, and probably better, approach notes that "connectivity" is critical to workstations, but not necessarily to PCs. We will treat the two as being almost equivalent, but with recognition that workstations normally have more power, flexibility, connectivity, and a higher price than do PCs.

inadequate software capabilities are deemed responsible should be examined with careful attention to the full context in which the limits are arising.

There are many potential beneficiaries of strong growth in the markets for PCs and workstations. The most immediate ones are the manufacturers of hardware and software. However, the history of development and use of computers (and of most other major technologies) makes it clear that many other stakeholders will be involved, and will have their activities and capabilities affected by the level of use of PCs. As a more indirect, but still important, influence, even the way in which markets evolve (e.g., emphasizing products and services aimed at individual consumers, or at particular business or technical sectors in society) can affect the breadth of use of the equipment, the kinds of users involved, and the resources made available to individuals and society as a whole to support the creation and use of information. The nature of the resources made available, in turn, can influence the answer to a question with enormous potential social and business consequences in the longer term: What are the types of skills likely to become needed by individuals and organizations (and nations) if they are to survive and flourish in a period when PCs and workstations are in, essentially, universal use?

In this broader context, the relevant questions must have a corresponding scope. Thus, we would like to know:

- Are hardware and software manufacturers searching for an industry growth "driver" in the right place? In the right way? Have they defined the general problem broadly enough to match the reality of what is happening?

- Are new types of application software all that is needed to make the difference? Is the tremendous progress being made in most of the hardware end of the industry acting to hide some types of important hardware deficiencies? If so, what role do these inadequacies now play in limiting growth? What about other inadequacies, such as the limited external information materials that currently are available to most users in electronic format?

- To what extent does the growth decline at the turn of the '90s derive from user limitations – that is, from the limited skills, interests, and time "available" for learning on the part of current and potential new users? To what extent do the difficulties derive from a mismatch between existing application functional capabilities and continuing user needs? Between user habits and computer operating requirements?

- How do the various pieces – hardware, software, and user characteristics – fit together to determine what the computer industry can become? How do these same elements influence what the users, and individuals in general, *may have to become* to survive and prosper in the future?

1.2 Finding the Patterns of Change

While elements of all of the above questions are examined in this report, we particularly emphasize the last series – how the pieces fit together, and what the overall pattern implies for the future of individuals and industry participants. Since software development is central to our interests, we start (in chapter 2) by examining how programs for mass market applications have evolved – that is, how and why different versions have grown in size, scope, and capabilities over time.

It turns out that a specific pattern can be used to characterize this evolution, and to indicate in general terms what might be called the "natural paths" of development for an application. But carrying out the actions implicit in the growth pattern has depended on a parallel improvement in hardware capabilities. This improvement is needed in order to service simultaneously two general, but conflicting, needs: enhancement of the capabilities and scope of the individual application programs, and reduction in the complexity perceived by the program's existing and potential users.

The software creator's requirement to deal with a trend of increasing inherent complexity is one of the factors that sets limits on a program's coverage and capabilities at any given time. However, over the long term, the effects of these limits will tend to be quite weak. Helped by steady hardware advances, the ability to handle increased complexity also is growing; and programs for most major applications areas have natural growth paths that will continue to permit their improvement and expansion for the foreseeable future.

While an understanding of how software packages have evolved can give us a sense of their possible future growth paths, it does little to suggest new application areas. Thus, it has little to say about how the general use of PCs may change over time. To cover this aspect, we turn (in chapter 3) to the activities that generate the needs that computer programs seek to satisfy – the processes involved in using information. Here again, it is possible to develop a generic pattern of what goes on: we can define a set of processes and subprocesses that must be employed, in some order, whenever one seeks to benefit from information. From this, we can note that

- For a variety of institutional, technical, and economic reasons, PCs are used now to support only some of the different types of processing activities central to the use of information.
- Quite specific possibilities exist for a family of related applications, based mostly on extensions of current searching and matching techniques, that have potential value in many of the processing activities where PCs currently have little or no usage.
- The new types of applications, if implemented, could alter the roles of PCs and workstations. They would become more continuously involved with their user's work, requiring less frequent but more skillful user attention, and performing a controlled portion of tasks that now require human efforts.
- While many uncertainties exist, we already are in an excellent position to start testing the possibilities. These tests need not be expensive, and they could contribute immediate benefits to at least some users, while providing guidance for future experiments and new software designs.

The new types of software mentioned above can place new types of requirements on users. These requirements could lead to some long-term, and largely unanticipated, changes in the skills that become most important for using PCs effectively. Our findings (discussed in the final chapter) suggest that

- There are a variety of limited, but quite significant, information processing activities for which computers can be "trained" (i.e., dynamically programmed) to replace human intervention in assessing the meaning and value of the substance contained in specific information items.

- To provide the needed training, the user must have (in addition to an appropriate level of knowledge of computers and programs) a deep understanding of language — of its capabilities, its limitations, and its local usage. In effect, the language requirement is not far different from the definition of what was needed to be fully literate in a classical sense.

If this situation arises, it will result from the growth of computer capabilities to where, for at least some (carefully monitored) purposes, they can manipulate words and phrases in a way that will give an outward appearance that they understand the symbolic content that the words and phrases are intended to convey. Given the fundamental level at which this type of change would be taking place, many other impacts are likely to follow.

CHAPTER TWO

THE NATURE OF SOFTWARE EVOLUTION: MANAGING COMPLEXITY THROUGH BUNDLING

2.1 Evolution of a PC Software Application

2.1.1 The robe of software

By 1989, some of the more popular PC software programs were being published in their fifth or sixth major versions, with most of the newer versions incorporating what their designers regarded as significant enhancements in performance over their predecessors. A review of the history of these changes can provide a sense of how successful types of PC software have evolved. The pattern of evolution, then, can give us clues concerning some of the underlying forces that influence the markets for PC software, and the capabilities these packages must offer users if they are to be successful competitors.

Because of its ubiquity, the word processing function furnishes a good example for this type of examination. Even before PCs came onto the scene, the groundwork for this application had been laid by a series of important advances in how typed materials were prepared. The earliest major advance of the modern era was the introduction of electric-powered typewriters. Improved over time in many ways, two of the most important changes were the development of the "golf ball" typing head (which, among other benefits, provided an easy means to change type fonts) and the incorporation of a capability for direct use of a "white-out" correcting tape.

Typing equipment that incorporated electronic capabilities entered the market in the form of specialized equipment, rather than as an application for a general purpose computer. The first commercial example of such equipment was a relatively large and clumsy IBM tape-based typewriter, later supplanted by a more flexible unit that used removable magnetic cards. Somewhat thereafter, Wang produced the first of its series of highly successful, minicomputer-supported typing stations. Meanwhile, but during the same general period, a variety of stand-alone electronic typewriters with removable storage media were marketed, such as those put out by the Savin and Quix Corporations. A number of devices with more limited capabilities (such as IBM's Memory Typewriter, which could handle corrections but lacked removable storage media) were offered, but none of these achieved any real success. The net effect of the earlier advances was that, when the PC emerged, the market had been sensitized to the use of electronic typing equipment. However, since most of it had been fairly expensive to acquire, only limited market penetration had been achieved.

Early PCs provided a weak technology base for supporting word processing. Their limited internal memory, in particular, placed strong constraints on how software designers could incorporate the functionality needed for their products to be attractive in the marketplace. Equipment limits often forced them to use complicated or clumsy means to accomplish what we now would regard as easy tasks.[2]

A relatively simple, yet dramatic, illustration of the magnitude of the changes in word processing software is given by the size of the programs provided to purchasers. A successful package in 1981, Apple Writer II required 48KB of random access memory to operate; and it was delivered on a single floppy disk, with less than 70KB loaded on the disk. Seven years later, in 1988, Version 5.0 of the popular WordPerfect was released.[3] It called for having at least 384KB of free random access memory; and the user package contained 12 floppies, holding some 4500KB of installation, conversion, program, and program support code.

---

[2] For example, the first truly successful PC (early versions of the Apple II) had neither a formal "Shift" key nor cursor controls. The 1981 version of the instruction manual for Apple Writer II recommended getting a hardware "Shift-key Modification," without which the "ESC" key had to be used as a Shift key. To control movement of a cursor within a body of text, a group of the regular letter keys were used, with the "ESC" key pushed twice to make a transition between text entry mode and cursor move mode.

[3] For all mentions of the WordPerfect 5.0 package, we will be referring to the IBM compatible version.

Where and why did this tremendous growth in program size arise? An examination of the functions offered over time by some of the major types of personal computer programs suggests that there are at least four general directions in which PC software packages can be enhanced by improvements or extensions. While there can be considerable overlap, in the sense that a single improvement sometimes makes contributions to growth in several directions, each direction can be characterized quite distinctly by its own specific purposes. And there are competitive reasons for all of the directions to receive attention. Using word processing packages as an example, growth occurred in the ways described below.

2.1.2 Directions of growth

The first direction might be termed a downward direction, since it leads to greater "depth" for the application in the sense of increasing the *variety, detail, and ease of use* of both general purpose and specialized "typing" features. Over time, improvements in this growth direction have included the following: easy multiple-column typing; more freedom in the use of headers, footers, footnotes and endnotes; the ability to make "global" changes in words or phrases; more powerful macro instruction capabilities; more characters and special notation capabilities; finer format and printer control (including, for example, the ability to print sideways); and a host of others.

Associated with many of the above features are a set of more general measures, designed to simplify user operation. Some typical examples are simpler and more intuitive menu and command structures, and movement towards "what you see is what you get" (WYSIWYG) displays.[4] These types of improvements ease the entry of new users and lower training and

[4] Measures intended to simplify essentially the full range of a user's work could be considered a separate direction of growth. However, because of their potential use with many types of applications, it probably is best to incorporate the structure for the more basic ones (such as WYSIWIG capabilities) in the computer's operating system (as was done for the Macintosh), or in a special purpose program (like Windows), rather than developing the full capability independently for each individual application.

updating costs; thus, they act directly to broaden the potential market for the application.

The second direction can be described as an upwards one, since it points towards *vertical integration*. This path leads to incorporation of page composition and document publishing techniques, and to improved communications capabilities. It involves offering means for varying fonts, character sizes, and other character attributes almost at will; easy exporting and importing of tabular and image materials, as well as normal text; networking with other computers and systems; flexible "boxing" of items of all types; and similar formatting features. Here again, WYSIWYG capabilities are very important for expanding the potential market, since they offer a tempting route for lightly-trained users to improve their skills while in the act of producing useful output. When a word processing package has incorporated these tools, the program is in a position to compete with other programs for a larger fraction of the tasks associated with first desktop (informal), and then in-house (more professional), publishing.

The last two directions of growth can best be defined as "sideways." The third direction involves a kind of conglomeration – the *incorporation of associated, but previously independent, activities*. Examples for word processing include spelling checkers, thesauruses, hyphenation dictionaries, grammar and style guides, improved search and retrieval programs, outline builders, and basic spreadsheet capabilities. The incorporation process generally makes the user's work easier and often improves the output quality level; in these ways, it can add considerably to the competitive value of the package.

The fourth direction is devoted to maintaining competitive position in the face of major technical and market changes. At this stage in the development of computers, these changes have operated to increase the *number and/or complexity of interfaces* with the external environment that are needed if an application package is to be successful. A good example here is in the area of printer interfaces. The 1981 Apple Writer II program had only a single, general purpose printer "driver."

Printer vendors were expected to provide either complete compatibility with this driver or a means whereby users could build their own software interfaces. In contrast, early versions of WordPerfect 5.0 could handle some 140 printers – and the number keeps growing. Further, since the newer printers have considerable built-in computing power of their own, their drivers must be larger and more complex if their capabilities are to be exploited fully and easily. Similar remarks apply, in varying degrees, to other major peripherals, such as display monitors.

2.1.3 Sources of complexity

To a certain extent, every important advance in computer hardware will lead to more complexity[5] in software. For competitive reasons, software designers have to exploit hardware advances to improve functionality. Also, if they expect to stay in business, they must protect existing customers from negative consequences that may arise from the changes they are making. At a minimum, facilitating change requires that they provide at least some upwards compatibility: each new version of a particular software package, at the price of being made somewhat larger and more complex, must be able to work with output produced using earlier releases of that program. This feature makes it easier for existing customers to upgrade from older versions of the program; and it can give confidence to potential customers that they won't be dealing with a vendor that might, at some future date, leave

---

[5] Since the term "complexity" is used frequently throughout this paper, a definition may be helpful. An item is *inherently complex* if it is composed of a large number of parts that are intricately combined. That item will be *perceived* as complex if problems arise, due to choices made during design and/or construction, that require users to understand many of the details of this composition before they can use the item effectively. Design can be used to introduce simplicity in application by hiding the complexity from the user, but this process has two cost elements. First, the designer must handle the extra complexity of the simplification process. Second, unless all important uses of the item can be anticipated in advance (or measures are taken to allow selective access to the hidden details), there will be desired operations that would have been possible in the fully complex version but have been lost to the user by having been made invisible.

them with files and other materials that have become useless or awkward to employ.[6,7]

The directions that evolution can take, and the motivations encouraging each of them, are shown in **Figure 1-A**. Establishing, for a given application, just how much growth has taken place in each direction would not be an easy task. Definitions of what do and what do not constitute features to be considered in each category must sometimes be arbitrary and ambiguous; and, as already noted, many features can contribute to growth in several directions simultaneously. Even if these problems were solved, rather complete knowledge of the programming details of each package also would be needed. After that, there still remains a big step to translate the programming details into an acceptable measure of effort or complexity.

Given the many difficulties described above, we have not devoted any serious effort to a quantitative analysis of the growth pattern of word processing programs. However, to give a feel for some of the numbers involved, we have prepared the material shown in **Figure 1-B**. Here, we once again use Apple Writer II as our starting point, and WordPerfect 5.0 as a typical example of the better 1988 products. All material was taken directly from manuals, or from examination of listings of files on the diskettes provided to purchasers. Some aspects of improvements along the vertical axis are hidden by this approach because, in the later application versions, they have been integrated into the main programs. Therefore, they do not get listed as identifiable, separate files.

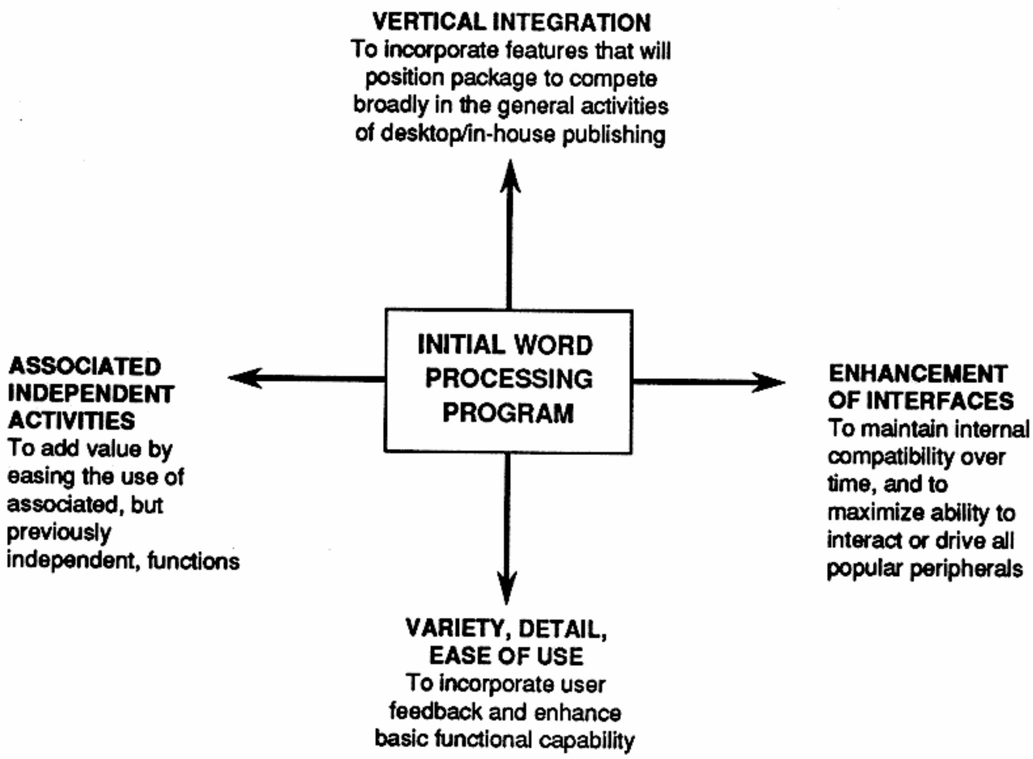
---

[6] Conversion capabilities also may be devoted to making it easier to switch "brands." Thus, a spring 1990 mail-order solicitation from Borland reads: "[Y]ou won't lose your investment in your current spreadsheet files. QUATTRO PRO lets you read, write and load even the largest files created with 1-2-3 Releases 1A and 2.01. . . ."

[7] Not all steps to improve a program's interface capabilities are taken for the benefit of users. For example, some programs have been written in higher level languages that slow the program's operation (to the presumed disadvantage of the user) for the sake of improving the market position of the program, by making it easier to adapt to different types of computers and operating systems.

Figure 1
Evolution of the Word Processing Application

A. DIRECTIONS AND MOTIVATIONS



B. PROGRAMMING EFFORT DEVOTED TO GROWTH DIRECTIONS
(in terms of KB of software provided to purchasers of application package)

DIRECTION & ITEM	APPLE WRITE II (Total = 70KB)	WP 5.0 (Total = 4500KB)
VERTICAL INTEGRATION		
Fonts and Graphic Support	About 2KB	272KB
Other	None	Considerable, but not measurable
VARIETY, DETAIL, EASE OF USE		
Various	About 6KB	Very extensive, but not measurable
ASSOC. INDEPENDENT ACTIVITIES		
Spelling Checker	Independent, stand-alone program; works only on upper case text	330KB
Thesaurus, Hyphenation Dictionary	None	708KB
Other	None	Considerable, but not measurable
ENHANCEMENT OF INTERFACES		
Printer Selection and Control	About 8KB	1660KB
Conversion	None	311KB
TUTORIALS & "HELP"		
Tutorials	About 7KB	375KB
"Help"	About 2KB	102KB

The growth patterns of other types of applications, such as spreadsheets and database management systems, have many similarities with the word processing pattern. The potential directions of growth appear to be almost universal, but emphasis among them varies with the starting point offered by a particular application. In all cases, the trend has been for dramatically large and rapid increases in program size and equipment requirements. The growth in requirements has provided a form of positive feedback, making obsolescent a lot of user hardware, as well as software. It also has modified the procedures and management techniques needed to produce current application software; the "garage-built" programs that characterized the early days of personal computers have pretty well disappeared from the mainstream. Finally, the pattern of continuing changes in application capabilities has had a variety of influences on prospective end-users, and especially on their decisions about whether and when to buy new hardware or software as well as what types to purchase.

2.2 Why Complexity Keeps Growing

As major application programs evolved, their developers and users faced quite different situations. The developers had strong competitive pressures to grow in functionality and interface coverage, and a simultaneous need to make their products easier to understand and use. The responses to all of these forces added complexity to the designer's job (although periodic consolidations of minor improvements often offered opportunities to simplify program or subroutine architectures). The users, meanwhile, have had the problem of building and maintaining suitable levels of skill in applying the application programs important to their work or interests. For specialized workers (e.g., typists and data entry clerks), the expected skill levels normally were high, and formal training sessions usually were provided by schools, employers, and/or vendors. For other users, expectations were lower. Here, organized training often has been widely available (though sometimes only at direct cost to the users), but it has tended to be more general in nature and more difficult to employ effectively. However, given an application of broad utility, there still are incentives to develop skills – users can derive a variety of benefits from having mastery over the majority of the program's capabilities.

2.2.1 Users and their problems

Even the most sophisticated group of users never employ much of the huge amount of code supplied with a program. A considerable portion is designed for use only when installing the application, or when adding or replacing equipment. Even here, much of the code will concern equipment types that the user doesn't have and likely never will have. The portions of code of continuing relevance to individual users is probably well under 50 percent of the total; and it may be as little as 10 to 20 percent in cases where the user makes no effort to exploit more than the bare useful minimum of the program's capabilities.

In spite of low utilization of program capabilities, the *nonspecialist* user faces significant learning problems in all except the "very minimum use" case. To illustrate, the quick reference card for WordPerfect 5.0 lists the keystrokes for some 235 "Features" (plus 18 strokes for Cursor Control operations).[8] These keystrokes cover three, and in a few cases, four levels of menus; at the final level, they still may require entry of selection choices or, very occasionally (such as when creating a macro), entry of "strings" of characters. Although skilled use does not necessarily require an understanding of all the features, a reasonable exploitation of the package's value might involve

- An understanding of one-half to two-thirds of the features; this level both permits effective use of the program and provides a good base for rapidly updating or extending knowledge of how to use the program, if either becomes necessary.

- Ability to use (and, preferably, also create for personal use) some ten to thirty macros (each with its own keyboard entry code).

- Considerable speed and dexterity at both keyboard entry and cursor (or mouse) control.

---

[8] A somewhat larger number of features is shown on the "Help" screens; these offer more detail and include about 500 items.

When this type of learning requirement is extended from one application area to, say, three principal ones and three to five smaller programs (some of which may be used infrequently, but can be very important when used), the full scale of the commitment that the user must make to be effective with a PC/workstation begins to emerge. To complete the picture, additional learning requirements owing to the rapid rate of change of programs already in use, and from the gradual addition of new programs (and/or replacement of specific current makes with other ones), also must be considered.

What kind of person is this "general purpose" PC user, who faces such massive learning problems? Here also, major changes have taken place. The expansion of applications has been paralleled by an expansion in their markets – a huge growth in size, accompanied by comparable changes in the levels and nature of interest of users with regard to computers and software. The early, circa 1980, PC users (other than specialized personnel, who would be trained either in schools or by their organizations) had to be part hobbyists. Without considerable personal interest, they could not learn enough to make the struggle to use an application worthwhile. Major procedures, such as equipment hook up and print operations, often were extremely complicated, and the support documentation generally was quite weak.

The 1990 general purpose user is quite different. There has, in fact, been a kind of bifurcation:

- On one side, large numbers of (generally older) people have become users less through active choice than through necessity – by direction from their managers, or in response to (internal or external) competition or employer cutback in numbers of support personnel available.

- On the other hand, a new generation of users are being trained in high school or college (or, increasingly, earlier) in the use of computers – and they are accustomed both to thinking at a keyboard and to repeated learning of new types of software.

The first user group leads to a large population of "disinterested" users; that is, people who may have great interest in creating or manipulating the *outputs* of computer applications, but who mostly have little direct interest, ability, or comfort in the *processes* of learning how to use the applications. For them, in brief, the simpler and more intuitive a program, the better.

The second user group contains a population with much more of a feel for computers and software. A portion of them often will treat learning a new program as an enjoyable challenge. Even for this population, however, there are usually strong reasons to favor simplicity in most aspects of widely used applications. Like the first group, most of these users have a primary interest in results. Even among computer enthusiasts, an important reason for simplicity can be to conserve time and energy so that they can broaden their contacts and experience with a wider range of types of applications.

For both user groups, designers face the task of increasing the functionality and utility of their products, while simultaneously making them easier to use. And every step in these directions forces them to take on the burden of dealing with greater complexity so that their customers can be rid of that very burden.

2.2.2 Why complexity keeps growing

A massive amount of feedback drives the system:

1) Another level of complexity always is available to be programmed – another layer of detail, or functions to be integrated.

   - If there is a "Peter's Principle" of application evolution, it is that efforts at integration or increased detail always go just one step too far too soon. Ambition outstrips ability (and schedule and budget) so that the last step ends up a little too complicated to handle well. In this sense, the primary limit at a given time may well be set by the exhaustion of budget, builder, or user!

   - But even this limit usually is temporary. Improved hardware capabilities will offer scope for new changes, and continuing simplification (or stability) of earlier program routines eventually will release energy to deal effectively with a new level of complexity.

2) A reason always exists to want to be able to handle another level of complexity; if needs don't exist, they will be created.

- The most rational *user* reasons for seeking and/or accepting more complexity are competitive – to be able to offer better service to customers, or to execute a task faster, or more completely, or more elegantly than a competitor (internal or external) can.

- Another common, but often less rational, reason lies in the inherent bureaucratic nature of society. There is no way, for example, that our income tax returns would be as complicated as they are if we didn't have computers to facilitate both preparing and processing them. In fact, computers have had a considerable role in encouraging a race between the invention of complicated new tax avoidance instruments and procedures, for tax-averse wealthy individuals and organizations, and complicated new methods and forms for collecting taxes on behalf of all types of governments. Needless to say, the private sector has no shortage of similar types of examples.

- For these reasons, it is hard to go backwards. Murphy's Law of Application Evolution probably reads: "If you ever strip out a few 'lightly used' features to make a program smaller and simpler, you always find out that you stripped the wrong features." Features tend to sell, even if seldom used, so stripping provides too few benefits for the risks involved.

It is tempting to say: "That may be correct in the short term; but programs eventually mature and stabilize, and the marginal benefits of additions to a program eventually become smaller than their costs." Although undoubtedly correct, this statement must be viewed with great caution. The application itself[9] is unlikely to stabilize indefinitely, so, if a particular set of programs stops changing, new and more appropriate programs will emerge sooner or later. An equally important consideration is that the "rules" applicable to software publishing are very different from those of any previous form of published media. For example, because they can be made from high information density storage materials, software products have small weight and bulk; this enables

[9] It is important to distinguish between "applications" and "programs." A program is an information *product* (that is, a bundle of substance in a given format, made by a specific set of processes) designed to meet a *need* (the application) for a given type of user in a particular environment. The phenomenon of building bundles is quite widespread at all levels of information activity. For an expanded discussion, see: A.G. Oettinger, *The Information Evolution: Building Blocks and Bursting Bundles* (Cambridge, Mass.: Program on Information Resources Policy, Harvard University, 1989).

them to overcome most of the practical limits on the amount of substance incorporated that have constrained the sizes of paper products. And the marginal production cost for an additional copy of a program ready for installation is very low.

There are few inherent limits on absorbing the desirable capabilities of related, but initially quite independent, applications. Market sizes can become enormous, spurred in part by extra benefits that are perceived to flow from widespread usage of the same program. Even if a major application stabilizes for a while, and ceases to grow in any significant way, it can take considerable time before it reaches this stable period. However, in our specific case, if one takes seriously even a small fraction of the hopes invested in hypertext, hypermedia, automated animation, multimedia and others, operating in a world with widespread image scanners, facsimile, and public combined voice-data (and, eventually, video?) networks, who can say when the "word processing" application of the future will approach a period of stable maturity? Or what modes of communications, and types of processing functions it will encompass at that time?

2.3 The Uses and Limits of Bundling

To complete our review of the evolution of software, let us examine briefly the general technique that designers employ to surmount the complexity problems facing them. Fundamentally, the process is one of integration. It is implemented in the form of bundling: taking a collection of individual elements, or steps in a goal-oriented process, and organizing them together in some manner. In this case, the organization involves building a new type of structure that, among other things, hides from the user many of the details and complex relationships inherent to conducting the tasks that a computer application supports.

2.3.1 Language bundles

At the bottom, reality for the computer resides entirely in electronic impulses derived from machine language instructions. Above this lowest level is a series of higher level "languages" in the form of operating and utility systems, programming languages, and application programs. Each level uses and bundles parts of the lower level languages into a form that its developers believe can be used to perform a particular function or group of activities (such as writing an application program, using a programming language, or building a spreadsheet with an application program) faster and easier than was possible previously. The tremendous capabilities of the computer then handle the backwards "translation" process to machine instructions in a manner invisible to the user. Finally, at each level, it is very desirable that a means of access be offered to many of the hidden details. Offering this access provides a degree of flexibility, so that the more skilled users (normally developers at the next "higher" level, or, at the application level, professional end-users) can make trade-offs to better match the software to their particular needs.[10]

Implied above is an analogy between spoken/textual languages and formalized sets of computer instructions. The analogy is a powerful one and offers a broadly useful perspective. On this basis, all application packages are simply special languages for performing a set of functions. The process of learning a new application involves learning its vocabulary (the relationship between a command, whether given by keystroke or by manipulating a cursor or mouse, and the results that follow) and its grammar (organization of command complexes). Programs become easier to learn and are viewed as more "intuitive" if their structures match the logic of the task being done (as practiced by the user), the structure of the user's normal spoken/written language, and the "logic" of the body (where physical manipulations such as keystrokes are involved).

---

[10] These trade-offs, when made by skilled end-users, can furnish valuable feedback to developers concerning features to be added to their next basic version of an application.

Also, just as learning multiple languages in a single family (say, the Romance languages) tends to be easier than learning the same number from different families, any availability across application packages of common standards, conventions and frequently used commands eases the learning process. This factor clearly encourages developers to adopt integration and incorporation tactics; no one wants to learn a completely new language for each function to be performed. Equally obvious, possibilities for making life easier for users are limited whenever patent or copyright protection (without licensing requirements) is extended to cover features where commonality can be of significant help to the learning process.

Although language-type bundling is the major means used by application program developers for making complexity easier to handle, it is not the only means available. A popular variant is the Graphical User Interface (GUI), pioneered by Xerox (borrowing techniques from artificial intelligence research) and made popular by Apple in its Macintosh series of PCs. Bundling is still at the heart of the technique and, in fact, the GUI might well be termed a "graphics language." In essence, it permits the user to handle command codes almost entirely by using a "mouse" to manipulate menus and icons in simple, intuitive ways, while operating with a WYSIWYG display. (It is worth noting that these types of interfaces require large amounts of processing power -- indicating again the close interactions between hardware capabilities and software possibilities.)

There seems to be general agreement that learning is easier for people when applications have a graphical user interface (GUI), a factor which has been important in many environments. However, GUI is not without its disadvantages. For example, many operations are slower using GUI, so skilled users often prefer to employ command keystrokes (which usually are provided for at least some commands) in place of the mouse manipulation technique. This practice suggests that a carefully balanced combination of the two types of languages could offer important advantages. And even this combination is unlikely to be the end of the trail. In the fall of 1989, Wang introduced "Freestyle," an interface/communications system that supplements the functions offered by a GUI with capabilities to enter handwritten notes or symbols that can be coordinated with voice annotations and comments.[11] Since the voice, graphic, image, and scanning capabilities of computers are growing rapidly, the evolution of a variety of mixed text, voice, and graphical languages seems likely and desirable. And they may bring with them new and better ways for achieving a balance between user convenience and simplicity of operation (through bundling) and flexibility to adjust to a diversity of evolving needs (through capabilities for selective unbundling).

2.3.2 Limits

Finally, there is the question of limits: How far can bundling be employed to counter complexity? Here again, the language analogy carries over. The underlying, and unavoidable, linguistic dilemma was well expressed many years ago by W.V.O. Quine:

> In logical and mathematical systems either of two mutually antagonistic types of economy may be striven for, and each has its peculiar practical utility. On the one hand, we may seek economy of practical expression — ease and brevity in the statement of multifarious relations. This sort of economy calls usually for distinctive concise notations for a wealth of concepts. Second, however, and oppositely, we may seek economy in grammar and vocabulary; we may try to find a minimum of basic concepts such that, once a distinctive notation has been appropriated to each of them, it becomes possible to express any desired further concept by mere combination and iteration of our basic notations. This second sort of economy is impractical in one way, since a poverty in basic idioms tends to a necessary lengthening of discourse. But it is practical in another way: it greatly simplifies theoretical discourse *about* the language,

[11] Because of financial problems, Wang had to abandon Freestyle. However, the concepts have not been dropped, and other organizations are continuing to develop similar types of interfaces. Pen-based systems, in particular, are receiving a great deal of attention.

> through minimizing the terms and the forms of construction wherein the language consists.[12]

Existing computer languages have staked out different positions on the spectrum of possibilities described above. In fact, the issue even extends backwards into the hardware side of the industry. An example is the introduction, late in the 1980s, of Reduced Instruction Set Computer (RISC) microprocessors. By using a smaller "grammar and vocabulary," they seek to gain in speed more than they lose in the complexity of programming with them.

What is as yet unanswered, since in most respects it is new to linguistics, is the nature of a language that combines graphics and voice with text in the same medium – especially when the medium is highly *interactive* and subject to considerable personal adjustment by the individual user. That is, to what extent can (or will) graphic and/or voice techniques add a new dimension to the use of higher level computer languages as a means to deal with complexity? In particular, if advantages are taken of the possibilities for interactions between computer and user, can these possible new languages alter the practical impacts suggested by Quine's comments from those we are accustomed to in the use of traditional languages?

---

[12] Van Orman Quine, Willard, *From a Logical Point of View: 9 Logico-Philosophical Essays* (Cambridge: Harvard University Press, 1953), 26. The author is a Professor of Philosophy at Harvard University.

PART TWO

NEW FRONTIERS: KEEPING THE "PERSONAL" IN PERSONAL COMPUTERS

CHAPTER THREE

USING INFORMATION: WHERE THE PC DOES AND DOESN'T HELP

3.1 The Steps We Take to Make Information Useful

3.1.1 The applications dilemmas

Where do we go from here? The previous section described an evolutionary growth process that appears to be a natural one for most significant PC applications. However, the discussion of complexity that follows this description implied some conditions and limits to the growth process:

- The conditions apply mostly to individual major programs; they require that much of the "growth" potential (largely made possible by hardware improvements) be devoted to building a variety of types of user interfaces and to hiding, from users, an increasing fraction of the program's complexity. The goal here is to broaden the market by making the program both more valuable, and easier to use and later upgrade.

- The limits apply mainly across programs; they arise because users have constraints on the investments of time and effort they can afford to make in learning a number of different applications, and then maintaining their skills over time (even for those programs used infrequently) as well as periodically updating them.

The above material can provide some guidelines for program development, but it offers no suggestions as to where and how suitable new applications, especially major ones, might be identified. The prospects here are rather discouraging. Intensive searches for "growth stimulating" applications have been underway for many years, with results that recently led one observer to note:

> **There is no "killer application."** What's more, it's becoming increasingly clear there is no single application to drive PC sales the way spreadsheet, word processing, database, and desktop publishing applications did through the '80s. None – sorry. Neither image, nor multimedia.[13]

---

[13] *The Gray Sheet Computer Industry Report*, vol. 26, nos. 2 and 3 (Framingham, Mass.: International Data Corporation, September 21, 1990).

If this comment is correct, a major source of the past growth in the markets for PC-type equipment has disappeared. Lack of a killer application does not mean that the market has stopped growing, but it does suggest that growth is beginning to follow more traditional routes. It also suggests that earlier industry hopes of developing a major mass market for PCs have been abandoned or delayed.

Aside from replacement and general diffusion, there appear to be two main thrusts early in the 1990s: penetrating new business markets (especially smaller and specialized businesses) by developing applications designed specifically for their support; and competing for a portion of the applications that previously have required minicomputers, or even main frames. The workstation segment of the market already is well positioned on these paths, and in many areas the PC is following.[14]

The larger-computer displacement thrust can lead to major changes in how PCs are used. Success here likely will result in their being configured more often in networked, relatively centralized systems than they are presently; hence the intense interest in Local Area Networks (LANs) for PCs, and in communications and groupwork software. The PCs in these systems will tend to look and act more like intelligent terminals for larger machines than did PCs of the past. The context for their use will make the most sense if it is less concerned with giving individuals tools to help them improve their personal performance, and more devoted to increasing the effectiveness of group operations. The end effect is likely to be a deemphasizing of the "personalness" of the PC in most of its newer forms of usage.

This pattern may well be the most likely one of the future and, as a business proposition, it can be a very good one for many stakeholders. But the implied prospective downgrading of the personal element has an unsatisfying feel to it. Are there no alternatives? That is, are there

---

[14] For example, Michael Swavely (president, Compaq Computer Corp., North America) notes: "Now . . . PC CAD comprises three-fourths of all CAD/CAE unit sales" (*supra*, note 13).

really no prospects for broadly useful, new types of applications that could help increase the market for PCs, but in ways that continue to support personal growth in capabilities? Since the more obvious fields for new applications with mass market potential already have been well tilled, finding an answer to this question probably requires looking at the information we use, and how we use it, from a different perspective than that commonly employed.

The information use areas of greatest interest to us are those that involve information employed for business and relatively serious social (i.e., not purely recreational) purposes, whether at work or at home. Both the substance and the sources and communications media used for receiving these types of information are very diverse. **Table 1** illustrates this characteristic by showing some of the main information sources employed by business decision makers.[15] Information used for other serious purposes will tend to have rather similar structures for the types of sources employed, although the specific entries often will be different.

3.1.2 The use process

To understand better how we use these (and most other) types of information, we can examine briefly the general steps involved in the use process. We describe below a set of such steps. Although they are relatively simple in concept, their full application often is complicated by a need to take convoluted paths through them, often with many iterations of some portions of the steps. The steps comprise the following six activities:

1) <u>Acquire</u>: To come into possession of an information item in a condition ready for examination. Possession can arise from a user's prior positive action, such as a specific request, or through passive receipt because of the user's position, status or some other set of criteria – or even through accident or coincidence.

---

[15] This table is an extension of Figure 10 originally published in Oettinger's *The Information Evolution: Building Blocks and Bursting Bundles*.

Table 1

Decision Makers' Information Sources

	From Inside Sources	From Outside Sources	From Personal Knowledge
Formal Processes	• Management information systems • Briefings, reviews, and other meetings • Memos, reports, special studies, etc. • Personal inspections of activities	• Media • Letters from suppliers, customers, etc. • National Association of Manufacturers, Committee for Economic Development, Harvard Program on Information Resources Policy, etc. • Consultants • Market research firms	• Schooling • Training
Informal Processes	• "What do you think, Joe?" • "Pssst...Do you know that...?"	• Golf course • Cocktail parties • Other social and personal contacts	• Experience

2) <u>Review</u>: To gain, through examination, enough understanding of the identity and background of an acquired item, and of the essence of its substance, that the appropriate next processing step(s) for that item can be determined.

3) <u>Search and Retrieve</u>: Working either directly or through an intermediary to identify, locate, and arrange for acquisition of information substance that is relevant to one's interests or responsibilities, or that supports current (or prospective) operations on an information item being processed. The activity of this step can range from a highly selective search for a specific item, to browsing for general stimulation.[16]

4) <u>Operate</u>: To take actions that add value to, or derive value from, an information item. These actions can include one or more of the following activities: absorb the general substance of an information item; structure and/or enter substance into a particular format; convert data or materials to graphic/image form; edit, extend, and/or transfer part or all of received or

---

[16] Browsing can have value that goes beyond simple stimulation. For example, it is one of the very few ways we have for acquiring published "Unk-Unk" information – that is, new information substance of importance to us that we previously did not know even existed.

created substance; determine what sequence of other operations or processing should be applied to substance (which can include using the current item to support operations on the substance of other items, or taking measures to store and maintain the current item if it has substance of potential future value) and then carry out these operations or processing steps. Finally, and usually the most important activity of the lot, is establishing by various means the implications of the item's substance for current or possible future action steps of all kinds.

5) Communicate: To prepare substance (which can be old or new, original or modified, aggregated or disaggregated, extended or edited, etc.) for passing on to others. The substance can be packaged in any of a number of possible formats, for any of a variety of purposes; and it may be prepared for use in many locations and kinds of situations.

6) Discard/Divert: To remove an item from the application of any further information processing by a particular user/recipient. Except when the removal is performed by diversion of the item to another, more appropriate, user, the item is not likely to be maintained; thus, unless duplicated elsewhere, it may be removed from any type of future formal consideration.

The above process steps are quite broad in scope, and, before they can be tested or applied, it is useful to divide each of them into a set of substeps. Depending on the situation being encountered, the substeps will be conducted either as alternatives to one another or in a sequence determined by the user. The substeps we use are summarized in **Table 2**. To help put them into context, **Figure 2** illustrates the more common paths through the use process and some of the linkages among the various steps. A more complete description of the substeps, and examples of their application in a number of common activities, are provided in **Appendix A**.

3.2 How the PC Is Limited Now: The Tasks People Bring to It

In an earlier Program report,[17] we characterized the situations in which computer-driven displays were preferred over print media for the

---

[17] Ernst, Martin L., *Electronic-Print Competition: Determinants of the Potential for Major Change* (Cambridge, Mass.: Program on Information Resources Policy, Harvard University, 1989).

Table 2

Summary of Information Use Substeps

ACQUIRE

1. by UNCONTROLLED/PASSIVE RECEIPT
2. by "NORMAL"/ROUTINE RECEIPT
3. by RESPONSE TO CONTINUING REQUESTS
4. by RESPONSE TO SPECIAL, ONE-TIME REQUESTS

REVIEW

1. by MINIMAL EXAMINATION
2. by CASUAL EXAMINATION
3. by ACTIVE EXAMINATION
4. by SPECIAL EXAMINATION

SEARCH and RETRIEVE

1. when RIGOROUSLY DEFINED
2. when "FUZZILY" DEFINED
3. when RECOGNIZABLE-IF-FOUND
4. through BROWSING

OPERATE

1. involving GENERAL ABSORPTION OF SUBSTANCE
2. using ENTRY, CORRECTION, SELECTION, AND TRANSFER PROCEDURES
3. using GRAPHICS/IMAGE CREATION TECHNIQUES
4. using ORDERING or RANKING or OTHER TYPES OF STRUCTURING
5. using COMPUTATION or SIMULATION or OTHER ALGORITHMIC APPROACHES
6. using PATTERN IDENTIFICATION and RELATIONSHIP DEVELOPMENT TECHNIQUES
7. involving PATTERN CREATION
8. involving FILING, INDEXING, and STORING

COMMUNICATE

1. with TEXT and NUMBERS — computer input
2. with TEXT, NUMBERS, and GRAPHICS — restricted distribution
3. with TEXT, NUMBERS, and GRAPHICS — widespread distribution
4. by ORAL DELIVERY — one-to-one or one-to-few
5. by ORAL DELIVERY — one-to-many
6. by ORAL DELIVERY — highly interactive

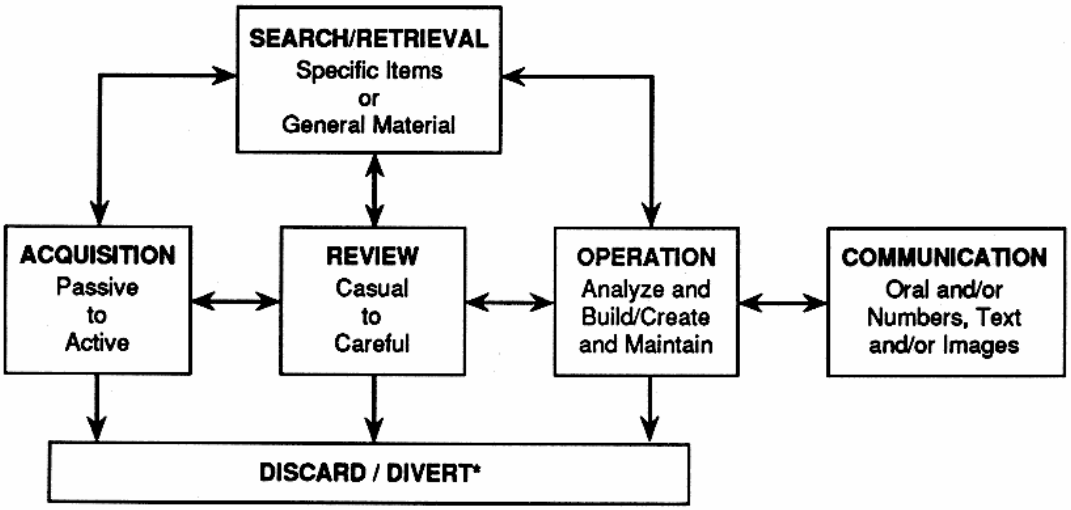
DISCARD

1. by DIVERSION — to alternative recipient
2 by "OUT BASKET" — for disposal by secretary
3. by TRASH BASKET — for disposal without intermediary review
4. by NATURAL PROCESS — such as disappearance in real time or user "departure" from source

Figure 2

Information Use Pattern: For Goal-Oriented Situations

A. GENERAL FLOW PATTERN



B. DETAILED FLOW RELATIONSHIPS

ACTIVITY	OUTPUTS to:	CAN INVOLVE or ITERATE with:
ACQUISITION	REVIEW DISCARD / DIVERT*	SEARCH and RETRIEVAL elements
REVIEW	OPERATION SEARCH and RETRIEVAL DISCARD / DIVERT*	Other REVIEW elements OPERATION elements SEARCH and RETRIEVAL elements
SEARCH AND RETRIEVAL	ACQUISITION REVIEW	REVIEW elements OPERATION elements
OPERATION	SEARCH and RETRIEVAL COMMUNICATION DISCARD / DIVERT*	Other OPERATION elements REVIEW elements
COMMUNICATION	OPERATION (Maintenance)	OPERATION elements

*DISCARD / DIVERT has no OUTPUTS or ITERATION partners.

presentation of information. We also examined both the underlying factors involved and the types of actions needed (whether currently practical or not) to enable major future growth in use of the electronic format. By reorganizing and extending the main conclusions of that report, we can develop a general sense of how PCs and workstations currently participate in each of the six information use steps described in **Table 2**.

3.2.1 Where we are at the turn of the 1990s

Many barriers exist that can prevent the use of personal computers: high costs, skill requirements, concerns about data security and integrity, fears of early hardware and/or software obsolescence, and numerous others. When PCs or workstations are employed in spite of these barriers, their use currently tends to be concentrated in office and home activities that meet at least several of five broad criteria:

1) The activities are well structured in a formal sense.

   This characteristic arises because suitable algorithms or other processing procedures for formal activities are more widely understood, and programming them is simpler and more effective, than for less formal functions.

2) The activities, especially those at home, are of a type that encourage user operation of the computer over relatively extended periods of time (that is, many minutes or more, rather than just a few minutes or less).

   This condition arises partly because most users (for a variety of reasons) are unwilling to leave PCs turned on but idle for extended periods of time. They are equally unwilling to accept the inconvenience associated with frequent, but separate, short uses – that is, repeatedly turning the equipment on and "booting" it, then accessing and using a particular program, and, shortly thereafter, securing and turning off the system. It also results, in part, from the need to make enough steady use of the computer to justify the necessary user investment in hardware, software, and skills development.

3) Performance of the activities requires either or both of these attributes: a) great timeliness and/or currency, and b) substan-

tial amounts of manipulation of text and/or numbers and/or graphics.

These attributes are those for which computers have the greatest inherent advantages relative to other available processing methods. In many work (and some home) situations, these attributes alone are sufficiently important to outweigh all the perceived disadvantages of using computers.

4) Except when the user's job is devoted to entry of data or other information, any extensive input materials needed for the activity either have been created by the user or are available to the user in PC-compatible form.

What this situation indicates is that few nonclerical/nonsecretarial users are prepared to employ an application where they must enter manually large amounts of material (other than that which they create) before the application can be used. There are some exceptions to this for data used in analyses and simulations, but very few for text entry.

5) Finally, except for activities where the inherent advantages of PCs and workstations (e.g., timeliness, currency, ease of remote access, manipulation capabilities, etc.) dominate user requirements, PCs and workstations seldom will be used for reading large amounts of text.

This situation derives from the generally low quality of electronic displays, relative to print, and the requirement that users adjust their positions and postures to the position of their equipment, rather than vice versa. Small amounts of text (such as abstracts and encyclopedia entries) may be read on computer displays, but large amounts usually call for a printed version of the material.

The effects of each criterion listed above can be changed in ways favorable to the future of PCs, although not all such changes are easy to make (or will, with certainty, be made):

- Coping better with criterion one, and, to a considerable extent, exploiting better criterion three, involves producing more applications for conducting less formal functions than most of those now in use. The descriptor "less formal" implies that the resulting programs normally should be easy to tailor to personal situations, needs, and preferences without demanding a high level of special user skills.

- Relief from criterion two involves overcoming two barriers. First, it requires implementing minor equipment and software design changes (to lower the risks and effective costs of leaving equipment turned on). Second, it requires establishing a greater range of popular uses for PCs (very likely including some of those "less formal" applications) to make it easier to justify investing in the equipment, and keeping it turned on.

- Criteria four and five are related, in the very practical sense that they both contribute to a situation where the range of substance available in an electronic format is inadequate to support some possible applications. Until a major improvement is made in the quality and comfort-in-use of computer displays, the types of materials presented on these displays will tend to be limited to items that do not require extensive reading. If a suitable display advance is made (such as development of a high quality, book-size flat panel display), then, *over time*, almost all kinds of materials can be expected to become more available in computer-compatible form.

3.2.2 Changes that could arise

With this background, we can review the current status and future prospects for PC applications in the various use steps described in **Table 2**. We will start by examining the steps one by one, and then consider briefly what the totality of possibilities add up to. At this point, we will move from history and conceptual development to speculation. The review of future prospects requires considering possible types of software that do not exist yet, so some imagination is needed. To keep in touch with reality, emphasis is placed on programs that might be created by extending or combining existing types, whenever this appears to be practical. Further, for new applications we concentrate on subjects where computers would conduct activities for us that we already are doing, with or without some kind of automated assistance. This approach is conservative, but experience suggests that widespread introduction of applications that create radical changes in information handling is a slow process, regardless of the merits of these applications.

Needless to say, all the specific programs described below are, at most, examples of possibilities.

Acquire. At present, users have few computer-based tools for managing their flow of incoming information, and most of these tools are operated by intermediaries, rather than by the end-users. The main mechanisms available apply to substeps 3 and 4 in **Table 2** (responses to Continuing and Special Requests). They include these activities: arranging and canceling subscriptions and special requests; sometimes creating and/or altering a selective dissemination profile; requesting a specific search for desired materials; and, if the user is in a senior position, influencing routing of materials and delegation of responsibilities for action. However, even these tools are relatively lightly employed, except in specialized work activities.

The immediate reason for the limited computer participation is that print-on-paper is the format of essentially all received items, other than some (largely internal) electronic mail. This condition offers few means for users to establish and apply useful flow management tools other than in fairly formal search-type situations.[18] Note that this limitation (i.e., limited materials received in electronic format) is a fundamental one that will apply, in some degree, to all the information use steps discussed below.

Since the basis for building any significant commercial markets has been relatively weak, little software is available that is relevant to the management of incoming information flows. There are available, however, some personal interest profile programs (for selective dissemination from news wires and electronic databases), and a variety

---

[18] There are some exceptions to this comment, for example, heavy users on special networks like INTERNET. However, even though INTERNET has perhaps a million users, the system's heavy users probably constitute only a small fraction of the total number of professional workers who depend on the availability of a variety of types of external information. These other workers either are not on the network or, if involved, they still receive their primary material mostly in printed form.

In contrast, originators of many types of materials have enormous control over who will acquire what, with growing abilities to be highly selective in exercising this control. Much of this control, exemplified by direct marketing and catalog sales activities, arises from use of computers to establish the detailed distribution decisions.

of automated full-text document indexing systems (used for searching electronic databases). Once enough materials begin moving in electronic format, these tools could provide a good starting point for developing user-controlled applications to filter out nonrelevant and redundant information, and to *encourage increased acquisitions of real interest*. Note that the latter aspect is particularly important; the system will make sense only if it improves the overall flow, which can be very difficult if filtering-out is the *only* operation.

Review. There are no computer tools to assist users in the Review step. Computer-prepared tables, graphics, color, and exception reporting (which, in some cases, may have been specified by users in senior positions) can help make materials easier to review, but decisions regarding these features usually are in the originator's hands, rather than the user's.

Once again, the proximate reason is the poor market potential, owing to the small amount of electronic material to be processed. Further, the background resources for developing applications in the Review area are weaker than those for Acquisition. A beginning might be made by pursuing the underlying concepts of exception reporting beyond the purely arithmetical versions now in use. Check lists could be established to call attention to certain specific features of an item (such as the possible violation of a particular planning assumption used earlier, or action of a particular type taken by a competitor). Similarly, specific (user-originated) questions could be asked whenever a user plans to discard an item. Another path would involve automatically prioritizing incoming items for review according to user criteria. The review step could be further supported by identifying and making available (or, at least referencing) materials already on hand that relate to an item about to come under review. Finally, some or all of the above capabilities might be integrated with the Acquisition flow control program(s) to form a single product class.

Note that user control should include the ability to exempt any desired items, by name or class (e.g., by source, subject, or size) from the automated Review process. Another key feature of all the above capabilities is that they should be dynamic – not only subject to direct control by the user, but also capable of automatically proposing a variety of adjustments on the basis of user feedback.

Search and Retrieve. Computers have a major and growing role in formal searches of the type shown as substep 1 (Rigorously Defined) of **Table 2**, and, to a lesser extent, substep 2 ("Fuzzily" Defined). However, they currently have no capabilities to help in searches of the substep 3 and 4 types (Recognized-if-Found, and Browsing), or for "informal" searches, as in **Table 1**, which often involve unorganized or nonelectronic materials. For these types of searches, the old-fashioned approaches of manual browsing or seeking human assistance are still the best means available.

Searches that employ computers commonly face these present limitations: the extent to which useful electronic databases are available in fields of interest to a user, the costs of building and/or accessing them, and the need for most users to operate through an intermediary because of the complexity of dealing with a variety of databases. A specific current deficiency is a general lack of useful internal electronic databases in most organizations and homes. Also, until recently, PCs didn't have software available that could index materials automatically, by a means such as full-text indexing, and then store the indices for later use in searches of user-created (or received and stored) items. This problem appears to have been solved, at least in part, by programs such as ZyIndex and AskSAM originally, and by Lotus's Magellan and, more recently, ON Technology Inc.'s ON Location.

As with several of the substeps, the main ingredients for change are the availability of more materials in electronic format, plus time to learn how to use them. The following elements could support or encourage the learning process: more common search protocols across databases (to lower the need to use intermediaries), increased development of organization internal databases, increased training at schools in the use of databases, and availability of low-cost, more effective scanners (to allow individual users to transfer more easily selected information from print to electronic format).

In general, no dramatic new software is needed, although considerable incremental growth appears both possible and desirable. For example, reading-oriented software (such as for browsing) will be required to assist in substep 3 and 4 type searches. Also, benefits can arise from incorporating normal improvements in search theory, as they arise, and from adding the ability to search for special features of items, such as specific references used, or the "density" with which tables of quantitative data, pictures, etc. are present. The main areas where advances might be encouraged actively are in efforts to use relational features in search operations, and to incorporate user feedback more explicitly during iterative searches.

Also note that, eventually, it will be important to extend these types of search capabilities to cover pictures and video; these areas may need major advances in search theory.

Operate. Operations capabilities are at the heart of computer activity – the primary source of its great strengths. This is where applications got started and where they continue to grow. However, even here they are not strong for all substeps.

- The greatest strengths are in substeps 2 through 5 (the Entry-Correction-Transfer family of activities, graphics creation, and the various qualitative and quantitative analysis techniques), where enormous amounts of software are available.

- With automatic full-text indexing, substep 8 (Filing, Indexing, and Storage) probably can be accomplished adequately for most personal file systems.

- Substep 1 (General Absorption) is weak because of the lack of electronic materials and the inadequacy of electronic displays for easy reading, as already noted.

- Finally, for subsets 6 and 7 (Pattern Identification and Pattern Creation), we lack an adequate understanding of the mental processes involved and, therefore, know not what to ask of the computer.

There are a number of directions in which steps can be taken to increase the role of PCs in Operations. Most of these steps are likely to arise over time as part of the normal growth process:

- Transferring technical applications from larger machines to PCs (as already is happening),
- Broadening the user base through a series of steps, to make it possible for less trained users to employ more complex analytical techniques, and
- Developing new types of programs that *may* help in going through the processes that individual users feel are important to their own analytical and creative activities.

Some examples of possible application areas along these lines are listed below:

- For qualitative general purpose and pattern-oriented techniques, provide a range of new tools, such as the following:
    - Improved, easily used list-building and manipulation techniques that emphasize identification of relational (including association-based) aspects of elements being analyzed;
    - Flexible graphic and sketchpad techniques that permit relationships and structures to be shown in a variety of (both two- and multidimensional) ways, and to be altered easily as thinking evolves;
    - Systems for rapid automated or semiautomated building and manipulation of association matrices of different types; and
    - Improved means for multiperson participation in qualitative problem sessions, ranging from more sophisticated forms of specially managed electronic bulletin boards, to online "brainstorming" support systems.[19]
- For quantitative procedures (including quantitative means to assist pattern recognition in its broadest sense), take the following actions:

---

[19] A considerable amount of literature and a number of software packages already exist that are concerned with computer-supported cooperative work. An example of a fairly large-scale 1990 experimental set of working installations is furnished by the Capture Centers, designed and operated by EDS, Inc.

- ▸ Simplify access and use of analysis programs, and provide guides and explanations of techniques with the programs;[20]

- ▸ Where feasible, perform automatic checks of the applicability/validity of techniques being used and of the significance of results being obtained; and

- ▸ Provide, when requested, "suggestions" for other techniques that might be of value.

- For preparing and maintaining materials for later use, ensure that the following options will be available:

  - ▸ Ability to easily strip off all, or portions, of received materials (with proper referencing and attribution attached) into personal work files;

  - ▸ Automated/semiautomated support for *personal* characterization (e.g., based on elements such as circumstances of acquisition, as well as on substance characteristics), summarizing (if desired), and indexing of all acquired items being stored; and

  - ▸ Future (low-cost) ability to transfer materials to new types of storage systems/media, if, or as, this becomes necessary.

Communicate. The capabilities of PCs in word processing and image creation enable them to participate in all types of communication, as the major vehicle, when creating text and tables, and as a key means for developing support material for the more formal types of Oral Delivery. Accordingly, PCs are strong in substeps 1-3 (all the text-based formats) and in 5 (oral delivery to large audiences). They have a lesser role in 4 (oral communications with a single or a few participants), but only in 6 (highly interactive group communications) have they been weak.[21]

---

[20] There are obvious limits to how far these explanations can go (and similar limits to the two items that follow). However, in some areas, like business statistics, a lot of misuse arises at the same time that opportunities are missed. We probably are at a stage where computer programs can help us cut down on both types of mistakes. If this type of advance seems far-fetched, recall that, not many years ago, Discounted Cash Flow calculations (now built into many pocket calculators) were viewed as requiring an almost professional math background to perform.

[21] There are, however, some beginning uses of PCs for this type of function; the Capture Centers mentioned in footnote 19 are designed to be useful in a wide range of informal meeting contexts.

Continuation of current trends can, by itself, overcome most of the present weaknesses of PCs in Communication. A few steps, however, could further strengthen their value:

- Encourage the migration and simplification of software that confers special advantages on the use of PCs, such as the ability to "zoom" and to perform 3-D rotations on appropriate forms of received materials.
- Improve the ease with which "lightly trained" users can create graphics of considerable complexity; eventually extend this capability to include useful forms of animation.
- Automate, for user review and concurrence, the production of indices, tentative summaries, and partial characterizations of all materials produced or (as desired) received.

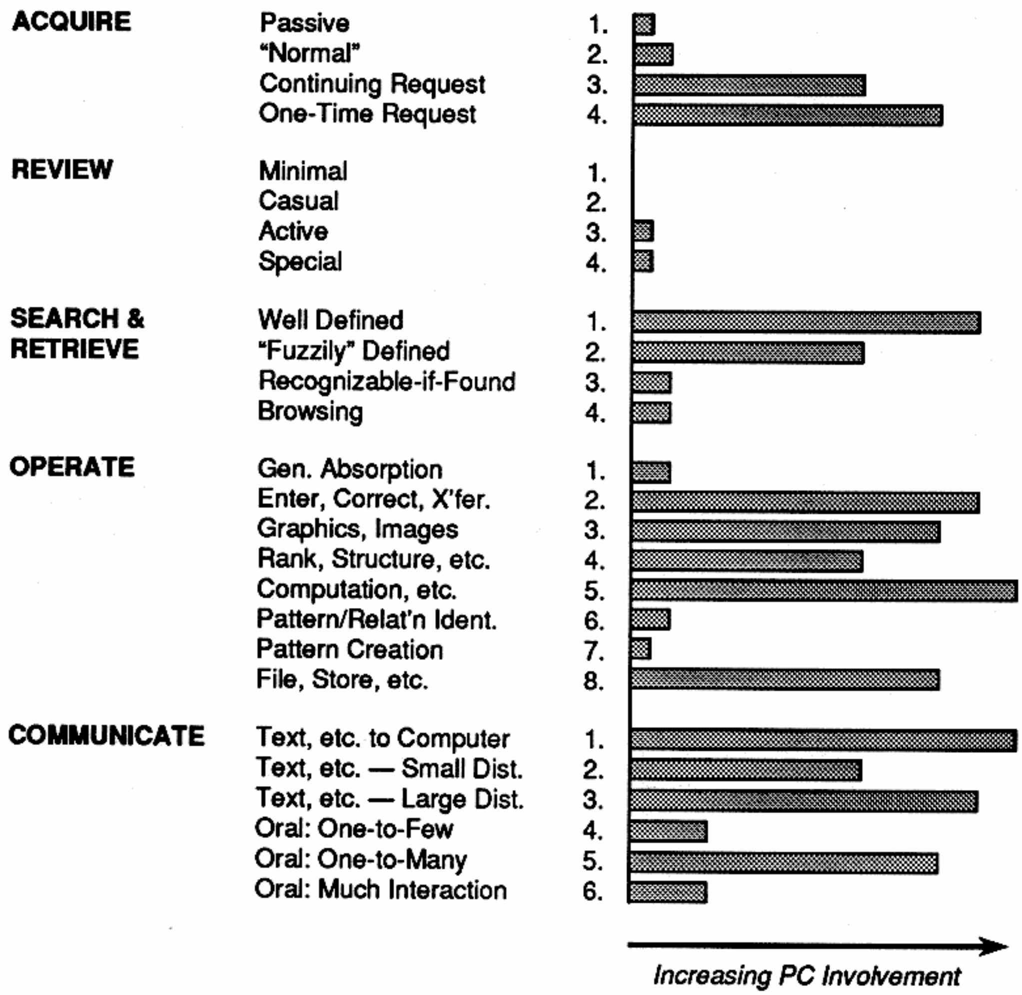
Discard. Discard normally does not require computer support (although deliberate destruction of computer records may need considerable user skills, especially as automatic backup systems of growing sophistication are developed). The most important change in this area *may* be a decrease in the need to discard as much personally received material as at present, on account of the continuing increases in computer storage capacity. A future pattern may arise where it becomes easy to store information in a "semilimbo" state – still accessible by special means if or when changes or new interests make some of these items potentially useful, but not cluttering up currently used files in the meantime. For the nearer term, however, there appear to be no special software needs for this activity.

3.3 Breaking the Limits, and Why It May Be Possible

The last few pages have presented what amounts to a special kind of "wish list." If we put all these pieces together, what do we have? **Figure 3** illustrates the general scope of the possibilities. It presents what amounts to a rough profile of the uses of PCs as we enter the 1990s, indicating the importance they have in the conduct of the different types of information process activities. The prospective new applications simply extrapolate somewhat the capabilities of PCs in areas where they already are strong (such as formal Search and

Figure 3

Profile of Personal Computers' Role in the Conduct of Information Use Activities — Turn of the 1990s



Retrieval, and much of Operations and Communication), and seek to build significant roles in most of the activities where they currently have little or no impact (principally Acquisition and Review, and the less formal modes of Search and Retrieval).

3.3.1 The texture of the changes

The picture of future possibilities that results is a rather odd one, with a different "texture" than we usually get when viewing how personal computers are used. To start with, instead of being used for a limited number of jobs, with each job requiring a fairly big "lump" of human and computer effort, the processes just described tend to involve a lot of rather small activities.

Further, the relationship between man and machine has somehow changed. For most current types of work where PCs are used, they are operated under close and continuing human control; for many of the tasks just described, the computer's work may well be done best if it is set up to operate much of the time at its own pace, with little active human control. A major part of its functioning might be spent in a quiet background mode, receiving information from one or more sources, and then processing it to make it more "digestible" (and, perhaps, more "tasty") for the human user. This type of partnership is new. *If it works*, once time has led to a suitable performance level (i.e., adequate "trust" has been established), a set of functions could evolve for which the human would serve increasingly to set context, scope, and policy, while the machine would carry out the desired operations largely on its own.

To make such a system useful, two characteristics must be built into it:

- The software employed for managing most of the key functions must be very flexible – that is, easily tailored by a user to match his or her (temporary, permanent, or a mix of both) situation, needs, and preferences.
- Second, some of this software must be able to do something much akin to "learning"; it must be able to receive various types of user reactions to an output, and then make adjustments based on this feedback. While other types of software will be needed, there must be a clear distinction between the "adjusting" and the "nonadjusting" types, with availability of the former critical to the new role of the PC.

Given these abilities, a radically new user environment could be built over time. From the user's point of view, it would represent what amounts to a new balance in the Quine type of dilemma, expressed this time in the form of the balance between simplicity in use and flexibility to meet different and changing user needs and skill levels.

3.3.2 Some practical considerations

At this point, several questions arise. Have we any basis for believing that such programs could be built? If built, would they be used? Is there a way to test the concepts? The uncertainties are great, but it so happens that there are several reasons to be modestly optimistic about our ability to learn the answers to these questions.

First, the concepts involved are not new. The long-term possibility of creating computer systems that could act as "personal assistants" has received frequent publicity during the last decade. Among the concepts proposed was one for "Knowbots," which would function specifically as personal "knowledge-handling" robots. By the turn of the 1990s, a variety of computer programs had been built that represent fairly serious efforts at information-filtering. Informal experiments by individuals appear to be fairly widespread. And, in very primitive form, some of the principles of the example programs already are well established in business operations.

An example of a commonly experienced existing application is a procedure used for handling questions about personal credit card matters. Some credit card organizations, using a voice-mail-type of response to a call, ask the users to punch in their credit card number and zip code before speaking to an agent. This system goes a long way towards providing automated Acquisition, partial Review, and nearly full Search and Retrieval for the impending Operations, *as viewed by the organization's representative*. The inquiring individuals have done all the work but are kept busy while awaiting an agent, and they probably receive better service overall. The organization obviously benefits from better personnel utilization. Although a trivial example, it is a real one that illustrates the principles involved at an elementary level.

Much more important are the possibilities for testing the example types of programs at a more sophisticated level. Much of the starting software for building the capabilities described in this section already exists. Some of the possible new programs fall well within the scope of the natural growth paths of current application packages. To a considerable degree, we already have the means to start testing, on a small scale, some of the potential applications. The fact that we already can test them *does not imply eventual success or inevitability;* there remain many other barriers to be overcome and some important questions to be answered, for example, about the practical value that the resulting products would have to users. It also seems clear that, even if successful, this will not be the much sought "killer" application. While it might develop considerable leverage over time, the application is likely to grow slowly at best, limited by the availability of relevant electronic materials and confined almost entirely to use in work activities. But this brief review does suggest that a little more analysis of possibilities and their implications is warranted.

CHAPTER FOUR

BRINGING THE PIECES TOGETHER

4.1 Turning Processes into Tasks

When the first digital computers were built, a huge backlog of work awaited them. For years, scientists and engineers had been developing theories, analytical procedures, and mathematical functions whose application required that enormous numbers of computations be made. The business and military communities had similar needs – computation-intensive problems or procedures whose solution could offer major economic and other benefits. Each increment of improvement in computer hardware and software, and each decrease in their costs, opened up another layer of practical opportunities to exploit these types of applications; and this process still continues.

The PC faced a similar situation when it came on the scene. We noted earlier in chapter 2 how predecessors of PC word processing already had started building a market for this application. Spreadsheets were in a similar state; they had long been recognized as highly valuable, but were limited in use by the labor required to perform the computations that gave them much of this value. Many database applications already had been implemented on computers, so the actions required here were more concerned with transfer than with development from a new and untested concept. An enormous number of other PC applications were developed over the years, covering at least some aspects of almost all significant areas of business and serious social activities. However, most applications that have been successful are characterized by a good match to a suitable subset of the five criteria described in section 3.2. That is, they have tended to involve discontinuous, well-defined formal tasks of considerable magnitude that require substantial manipulation and/or timeliness, but only limited reading of text.

The potential now exists for a suite of applications, illustrated by our examples, that go beyond the success criteria of the past. Further, the more important of these new types of programs would be devoted to activities that until now have not been considered appropriate for personal computers, except in special situations. Specifically, many of them would be devoted to what we now regard more as continuous processes than as discrete tasks, while the strengths of computers in the hands of individuals have resided in the latter area.

Can effective programs for these purposes actually be built? And, if built, are there reasons to believe that there will be real uses and a healthy demand for them? No complete answers can be given at this point in time, but some clarification of the questions is possible.

First, and most critical, the basis for most of the changes being discussed here *requires* that a much larger fraction than at present of all information made available to users be in *electronic*, rather than in *print-only*, format. As noted in chapter 3, attaining this level of availability of electronic information is likely to require a major improvement in computer-driven displays, as well as a number of other types of actions. However, market forces operate in both directions; if even limited success is achieved in some of the application areas that we have been describing, it could encourage and speed the rate of progress towards wider availability of the needed electronic materials.

Second, if effective versions of the exemplary new types of applications actually began to penetrate the market, a potential would arise for changes in the nature of PC/workstation involvement in a wide variety of information functions. In particular, PCs and workstations could become able to participate in information flow functions from the start – in fact, from before the user has even seen a new item of information – to the finish. This situation contrasts with their present role, where the computers are limited to a set of specific subtasks scattered throughout the process of handling a particular information item (or group of related items), but with gaps between many of these subtasks.

4.1.1 New types of searches

When looked at closely, the applications leading to this change are not as radical as they may first appear. We are all well acquainted with the use of computers to search for information once we get well into a job, and can define what we want. Of the example applications described earlier, a large fraction of those that introduce PCs/workstations into "new territory" are concerned with various types of searches. These searches are principally, but not solely, intended to help control and support the flow of information through the Acquisition and Review steps, which means that their scale is much smaller than that of most traditional computer-based searches. Instead of processing the enormous databases to be found in LEXIS or DIALOG, the activity normally would be confined to, say, a few tens of megabytes that are candidates for examination during a user's day of work. This should be a comfortable level of activity for a PC, even if the required processing is quite extensive.

On the other hand, the role played by these daily searches tends to make them somewhat more complex and less "sharp" than the ones we are familiar with now. While, in a technical sense, most of them do not require search procedures fundamentally different from those that we use currently, they do differ in at least three ways:

- While many, if not most, current searches are undertaken in an interactive mode, with successive search iterations narrowing in on the objective, many of the example searches only make sense if the PC can be "taught" to complete the job primarily by itself, with only infrequent user intervention.

- Although, in particular situations, one of the various types of searches described may prove worthwhile, in most cases it probably makes sense to proceed only if the expectation is to eventually implement a number of the search types.

- Finally, in contrast to typical (mainly literature-type) searches, the searches described here have many objectives and come in a variety of forms.

The types of searches that might be considered can be exemplified by a sampling of those discussed in chapter 3:

- Searching before we even receive an item, to determine whether we choose to receive and review that item or to divert or discard it.

- Searching positively – that is, continuously and more widely than we could do manually by ourselves, for information items likely to be relevant to a set of tasks or area of interests that we have defined.

- Searching selected classes of archives automatically, whenever our "interest profile" changes significantly, to help determine whether we wish to transfer some of these archives to more active status.

- Searching for "warnings," such as possible violations of specific planning assumptions, or data that suggest specific exception limits are about to be exceeded.

- Searching for a variety of indicators that can be used to determine the relative priority of attention that different incoming items should receive – again, *before* these items have started to receive our personal attention.

- Searching for related materials that can help either or both of, first, the review, and then the operations processes that follow receipt of an item. In a sense, this step simply provides us, at an earlier stage, with information of the type that we normally would seek later.

- Searching for suggestions of analytical techniques that might be useful, and for means to validate or assure the appropriateness of the techniques we have chosen to use.

The above examples suggest that new types of search software may provide one of the major means whereby PCs can expand their role in the use of information. Does this mean that software in this general area might come to act as the much sought new driver of PC markets? Possibly. However, it will take a lot of time and progress before this question can be answered. Most of us complain about having an information "overload"; but how many of us would really accept an offer to receive less material *even if we were sure* that the items eliminated would be only those with the least relevance to our responsibilities and our interests? And what would it take to convince us that a computer-based automatic system was removing *only the least relevant* items from our daily information flow? Access to information has long been important to survival and success, and old habits are given up slowly.[22]

What is much more certain at this point in time is that we presently have the means to start experimenting with and testing the possibilities in real life contexts at relatively low cost:

- Current PCs have the needed power and memory;

- Software like Lotus's Magellan[23] (for IBM PCs) and ON Technology's ON Location (for the Apple Macintosh) incorporate basic indexing, search, and other capabilities that can operate personal versions of many of the example types of search listed above;

- Existing personal (or unit) files, company electronic mail systems, network distributed information (such as over INTERNET), deliveries from commercial databases, and various other electronically distributed routine information materials provide adequate sets of *initial test data* within many organizations; and

- The macro instruction capabilities built into existing software probably allow users to create a loosely integrated test system, with a minimum of more complex programming effort required.

Once this type of testing has been started in a structured way, knowledge can begin to be acquired on what does and doesn't work and what is or isn't important.[24] This kind of feedback can provide the

---

[22] The reluctance to change old habits suggests that if users begin to employ computer information flow control programs, the first programs should be devoted to the least sensitive operations - for example, programs that establish the priority of items for attention, or that find and retrieve related items of information – rather than to operations whose programs would permit them to discard information materials.

[23] We make considerable use of Magellan as an example simply because PIRP uses the program for a number of purposes and, therefore, we have a reasonable understanding of it. Such references are not intended to "certify" this software in any way or to imply that other packages may or may not have similar, or even superior, capabilities.

[24] Quite a few informal tests of computer control of information flows have already been conducted. Some of these tests have led to products being offered on the market. One example is an experimental system called Information Lens, which was developed under T.W. Malone at the Sloan School of the Massachusetts Institute of Technology. This system can extract electronic mail messages that match user-specified

basis for starting both evaluation of potential market size and the design of more complete, formal versions of what might be described as information flow control and review support systems.

In summary, what may turn out to be the most important new ingredient for PCs and workstations, as identified during an examination of the detailed steps involved in using information, is a greatly enhanced capability to conduct a variety of searches of different types and for different purposes. Whether a software package offering these capabilities would be a market success – much less the long-sought new driver of PC markets – is not answerable at this point. However, a lot of economical, but practical, experimentation and testing of the concepts can be conducted with existing resources.

4.2 Fitting into the Traditional Pattern of Evolution

How do these considerations fit into the picture developed earlier, concerning the nature of software package evolution and the role of complexity in shaping the process?

4.2.1 Possible starting tools

Descriptions of the reasons why programs like Magellan are developed usually emphasize the difficulties caused users by the large (and rapidly growing) capacities of modern fixed disks. Information is lost because of the sheer volume of data being stored. Files are hard to find, since no simple systems exist that would help to easily describe their contents. Browsing is awkward and time consuming. Often, specific text *within* files is needed, and it is even more difficult to get at. One means to solve these difficulties is to develop a program designed to deal with them. A good indexing, search, and retrieval system is a critical component of any such program, especially if it is

---

interests and then sort them by purpose or priority. A program based on these concepts (Beyond Mail, from Beyond, Inc.) was announced in early 1991. Also, heavy users of INTERNET reportedly used tools like Magellan or ON Location to help manage items that have been sent to their personal electronic mailboxes.

important that specific pieces of text, or references to particular concepts or subjects, are to be identified and located.

A lot of major software applications incorporate search mechanisms. These usually are needed for rather narrow and specialized purposes. In word processing, for example, the most important use of the search function is to enable quick, easy, and controllable change of a word or a phrase, or its replacement by another throughout a document. In other applications, the object may be to rapidly locate specific files, data, or other information from carefully structured information banks. These types of searches have little in common with the systems for full-text indexing and interactive search personified by major database services like LEXIS/NEXIS and DIALOG – and replicable for PCs, on a much smaller scale, by Magellan, ON Location, and others. It certainly is possible that developers of a word processing or database application might decide to expand their search capabilities until they become a major component of their product. However, this action probably would be viewed as a diversion from their natural growth path, and makes the prospect rather unlikely.

Although the key feature of Magellan is its indexing, search, and retrieval capabilities, it generally is classified as a utility program. The search component provides a rather natural base from which to offer a variety of functions that help to organize, update, alter, and control directories and files, and to perform a variety of other management and housekeeping duties. One of the many features offered is the ability to move easily, rapidly, and efficiently from anywhere within Magellan into any other desired application, and then to return, with equal ease, speed, and efficiency, to the original location within Magellan.[25] A

[25] Any files written in ASCII (as well as the non-ASCII files from a growing number of popular software packages) can be viewed, *but not altered*, from within Magellan. Since the viewing may lead to an interest in changing or working with a file, the ability to quickly enter the originating program would be quite valuable. Hence, the motive for Magellan's designers to incorporate an easy means of access to the originating programs. The capability, however, is not unique to Magellan; it is found in other programs, such as Windows.

second important characteristic is that the program is easily "personalized" by modifying a set of "Dialog boxes" (which define how some of the main functions are to be carried out), and through use of Macros. In combination, these resources help give the program an excellent inherent capability to operate as a "Software Resource Manager" – that is, as an aid for reviewing, organizing, and otherwise managing data files as well as a "container" of, or starting point for using, a whole suite of different software packages. These features provide a valuable base on which to build capabilities for improved information flow management, discussed earlier in chapters 3 and 4.

Quite a few software programs have some, but not all, of the capabilities described above. MS-DOS Shell, Microsoft Windows, and Norton Desktop are probably the best known examples, although they all currently lack the critical indexing and search capabilities – which, in turn, are available in a number of independent programs that lack the other management features. The need for the full set of functions appears to be growing, but the programs offering them are not yet treated as constituting a specific, recognized class of programs and don't currently have a formal name as a group. Further, opinions differ as to how and where the required functions should be handled in the longer term.

Some professionals regard operating systems as the natural home for these functions. They consider this location to be the best for achieving what they term "seamless" data level integration, and consistent user interfaces, across all the programs available on a network. To some extent, PC operating systems currently are moving in this direction, following a growth path that emphasizes the following: more flexible user interfaces (including more support of WYSIWYG displays); better communications capabilities; and much greater ability to move easily among, and to selectively link, different application programs. However, with all the other hardware and software activities that operating systems must handle, there may be advantages to keeping most software resource management functions separate – especially for users involved with no more than simple or limited network operations.

For example, if the development is kept within a smaller scope program, it is likely to be easier for the software management function to evolve more rapidly towards meeting the needs of the "less-professional" users, who form the large majority of the long-term potential market.

Regardless of which particular kind of program takes over the task, it seems clear that a major new PC management tool is needed. Further, a major component of any tool designed to handle this need is a powerful indexing, search, and retrieval capability. Since several existing software packages already offer these capabilities in a fairly flexible application environment, it appears that the basic tools required for testing the types of applications described in section 3 have been largely developed already. Finally, creation of a complete, formal information flow control system might well lie directly on the potential path of development of this new type of generic software, which might be referred to as Software Resource Managers.

4.2.2 New requirements

This evolutionary pattern offers some hope for both new PC uses (and, as a result, new markets) and an ability to retain the personal image with which the PC started. However, it does little to change the complexity situation. Training requirements for many applications may decline, as training methods and materials improve, and as the applications mature and stabilize. Better and more intuitive user interfaces can ease the training load on a broad basis (and even make some of the training fun). In the opposite direction, the applications and environment of the future inherently may be more complex and sophisticated than at present; that is, it may involve more advanced analytical techniques, offer more options to be considered in word processing and publication activities, require more (and more clever) graphics, and so on.

Supplementing all of these matters, the very personalization features that can make a PC information flow control system so potentially valuable to individual users place their own significant requirements for skills on these users. Further, these skills go well beyond just computer knowledge. To illustrate, Magellan will perform searches on "synonyms" of a specified search term. These synonyms come not from a supplied dictionary but from a list compiled by the user. And the words listed need not be synonyms in the traditional sense; they can be *any* words the user selects. This ability to personalize the list allows the user to conduct a form of associative search – a potentially valuable tool, but one that is highly dependent on the skill with which the "synonym" list is built. In many respects, this new skill requirement could make for a new ball game – although, as described later in this chapter, it turns out to be mostly a very old ball game in a new guise.

4.3 Winners and Losers, but Mostly Winners

It is far too soon to start a detailed determination of who stands to gain or lose from the kinds of changes we have been discussing. However, some central themes in the change pattern have enough internal consistency and historical analogs to merit a few observations.

Partnership trend. The changes in the nature of PC use discussed here, and in the characteristics of the new software needed to support these changes, are consistent with a long-term trend that indicates computers of all types have been moving from being simply tools to becoming what amounts to "junior partners" of the user(s).

This move towards partnership requires a change in how key PC software is used. Currently, almost all PC operations take place only when the computer is under what amounts to continuous observation by the user. For the future partnership to work, PCs must have the capability to operate much more autonomously (although within a limited domain), subject to only periodic user review. In a broad sense, this capability also is part of a long-term trend, but it has been hidden by excitement over the opportunities, made practical for the first time by the PC, for users to have direct and close interactions with digital computer equipment.

Software products. If the changes start to occur, software producers should benefit in many ways. To start, the changes would imply that reading support software has come into use and encourage further development of such software. Success of the changes also depends quite critically on wider availability of electronic information, which suggests that expansion of the demand for electronic communications and delivery software must accompany any significant progress. The basic search programs, of which there may be a wide variety of types designed to meet different purposes, provide other potential (general purpose and specialized) markets. There also would be reasons to develop many kinds of support and control tools, such as the following: special purpose synonym and/or association dictionaries; mechanisms to modify normal search procedures for particular purposes; automated aids to make it easier to test how well a given personalized system is actually working; and programs (or sub-routines) to handle a variety of new analytical tools that may become important.

Mini-Applications. In another direction, there is the potential for some items of information to be followed by the computer from their selection to be Acquired and Reviewed through completion of all Operations and Communications activities. This complete involvement is in contrast to the current use of computers, which is much more disjointed and includes applications being employed for only some processing steps. If the more complete coverage arises, it could open up opportunities to build a series of new mini-applications that both aid and exploit this feature of the work pattern. While history suggests that few of these mini-applications are likely to be of major importance, great variety can be anticipated, continuing the rather tumultuous tradition of this industry.[26]

Hardware. Another probable winner, which may benefit in several directions, is hardware. The most obvious is simply an expanded market. Another could be the chance to differentiate products by offering PCs

---

[26] This opening up of a new field of software opportunities might lead to a temporary reversal of a trend mentioned earlier, and encourage another generation of "garage-built" software.

with somewhat new architectures. For example, from the nature of the activities involved, it might eventually make sense to design work-related PCs as two-processor systems. One of the processors would be general purpose, as at present, while the other would be designed explicitly for conducting fast and efficient search operations, perhaps mostly in a background mode. In its simplest version, a precedent for this type of action is the current use of math coprocessors when it is important to conduct intensive calculations.

Publishers and providers. Publishers and electronic information creators/providers are also stand to make large gains, although they face a number of problems as well. The key issue areas involved, if the changes start to be successful, concern the protection of intellectual property; development of pricing policies that can build the market effectively, while avoiding excessive cannibalization of existing print products; and (for some types of products) finding ways to generate advertising revenues (or an equivalent) so that prices competitive with print can be offered to end-users.

A lot of progress can be made before some of these issue areas become acute. Internal and other electronic mail, access to existing bulletin boards and interest group networks, and access to some commercial databases can go a long way; combinations of these sources offer logical starting input streams for developing and testing practical applications.

In the longer run, the costs for accessing and retrieving information from commercial databases can become a problem for both organizations and individuals. A number of existing pricing schemes could perhaps be extended in use to help here. A traditional scheme is to offer lower prices for bulk acquisitions (including, in some cases, limited and specific redistribution rights). Another is an analog of the "summary-full article" pricing approach; it would involve a low charge for letting a computer scan an item, to determine whether it is of interest, and an additional charge if an actual acquisition is made thereafter.[27] In any event, the problem is not new. Over time, experience should provide a sense of the values being received, which should help in setting any needed acquisition policies.

Telecommunications. Also standing to make significant gains in traffic are telecommunications organizations. Some interesting pricing possibilities also exist in this area, such as extra-low charges for trickling information through to PCs (and other processing units) during periods of very low line usage (such as the early morning hours). Also, as in the case of information providers, transmission companies may benefit from a charging system that has very low rates for "exposure" to an information item, and a higher rate if the item is then acquired for further use.

Networked PCs. Finally, mention should be made concerning networked PCs. They have mostly been neglected for the sake of focussing on a quite different set of possibilities. This neglect should not obscure the fact that networking and groupwork provide the areas where the greatest near-term growth in PC use and new software offerings are likely, and these activities may end up dominating the longer-term future as well. There is no contradiction between these possibilities and the ones we have been reviewing. Particularly for large organizations, networked systems with software that enables groups of participants to communicate and work together on a task offer enormous benefits. However, there appear to be no inherent reasons why use of networked systems need inhibit the personalization mechanisms described in this report. If both these areas gain market support, the two can supplement each other with many mutual benefits.

---

[27] If the future PC can be trusted to select, for acquisition, only the more desirable and relevant of available materials, this type of pricing might enable it to manage its own acquisitions budget.

4.4 The Individual User: Old Requirements in a New Context

We now turn to the last stakeholder topic: What may these changes mean to individuals? The answer is not a simple one, since it involves several quite different considerations. First, there is the new relationship between user and machine. For the new relationship to be successful, the user must adopt a new role on the user that will require a different level and type of investment of time and skills than in the past. Second, as explained in more detail below, in this new environment there will be benefits that can be realized best only when users have a clear understanding of the full potential range and the details of their responsibilities and interests. These benefits also offer strong incentives for users to develop certain kinds of greater self-understanding.

Turning first to the user-machine relationship, until now it has been one where the normal user has had to learn enough about computers and software to be able to command the computer to take one or more of a variety of well-defined steps. Typically, some of these steps are small unit operations (such as entering text, data, action codes, or option choices), while others invoke broader instructions (such as performing a set of calculations, or printing or saving a file). In all cases, however, the nature of the outcome will be known in advance (assuming the command is appropriate and given correctly); and, since the details are essentially deterministic, they also can be at least estimated before the command is issued.

The new requirement placed on the user is much better described as being able to "teach" the machine how to conduct itself under conditions where the objectives may be clear, but the results of any given "lesson" are quite uncertain and cannot be estimated in advance. The teaching process, in fact, must be a continuous one, in part because user needs and interests will change over time, and in part because experience and feedback form the only sound basis for making the teaching effective. The reason for the new modus operandi is simply that a major portion of the new types of user instructions (e.g., the various "search" instructions) will be dependent on the effective use of an extensive vocabulary of "words" – symbols replete with multiple meanings, ambiguity, and dependence on context – instead of relying principally on the much neater numbers, the very limited vocabulary, and the algorithmic procedures that have dominated in the past.

In at least some respects, this is new territory. No matter what the user's starting point, considerable practice and user learning seem likely to be needed before a comfortable level of personal skills is acquired. Even past experience with search operations will not, by itself, be enough. During most of those searches, many of the associated difficulties were partly dealt with by conducting interactive, iterative operations to reach the final results. The nature of the processes discussed here will permit iterative searches, but, for the most part, they should be conducted autonomously by the computer with only periodic "new instructions" from the user.

In this rather special environment, the user learning/teaching processes should get off to a better start, and should more quickly converge on an acceptable level – *if* the user has a very clear sense of the full scope of his or her responsibilities and interests. The goal of the information flow management process is to broaden the subject matter searched, while simultaneously providing a more efficient filter of the value of items to be accepted. For this goal to be met, the sources, types, and substance of information that can provide the best support in meeting work and personal requirements must be understood with considerable clarity; only then can a good job be done at both determining the objectives and the content to be incorporated into successive lessons and evaluating the results being achieved.

Finally, to best exploit the possibilities, users can benefit greatly from understanding how they react to and then process information. What excites and stimulates them? How do they integrate bits of information? To what forms of presentation, and from what sources, do they respond best? This effort to comprehend certain personal reactions should not be confused with an attempt to develop a general theory of "how we think." Rather, it involves efforts by an individual to gradually acquire a sense of what works best, out of the available options, *for that particular individual*. This type of understanding has considerable value in the teaching process, especially for tasks like entering terms for associative search operations.

When we put all these pieces together, we find that, in terms of critical user skills, we have a deep combination of the new and the old. In the new category, the requirements related to computers and software have been expanded, if only by the presence of new types of software. The mechanical aspects of learning and operating this software may require some additional skills, but the overall requirements should not differ much in character from those used with existing types of programs. The impact of the "old," however, can be quite dramatic; while old in a cultural sense, much of it will be very new to the PC/workstation world because of the strong language component of the instructions that the user must employ to teach the computer.

In a book written shortly before PCs began to be popular, Robert Pattison commented that "literacy is foremost consciousness of the problems posed by language, and secondarily skill in the technologies, such as rhetoric and writing, by which this consciousness is expressed."[28] Clearly, whenever an effort is made to "train" a computer – using words, short phrases, and some simple parameters as the means for a computer first to identify (autonomously) materials of potential value to a user, and then organize and present them in specified ways – the "problems posed by language" will be encountered head-on. As long as computers dealt with numbers, algorithms, and simple recordings of human inputs (as in data entry and word processing), the more fundamental problems of language and meaning could be finessed. To take the next step, if it proves to be practical, requires facing these problems more directly and in greater depth than has ever been required in the past.

---

[28] Pattison, Robert, *On Literacy* (New York, N.Y.: Oxford University Press, Inc., 1982), vi.

Of course, all people will not be equally subject to requirements to have, or have equal opportunities to acquire, these "new-old" skills any more than all people in the past had equal needs for, or access to, the "higher" levels of literacy. Where they are needed, however, the requirements are not trivial. Consider, for example, the classic problem encountered in all types of search: what "instructing words and phrases" (including words and phrases used for their associative value), organized in what particular manner, will best serve to ensure that searches in a given area are precise (i.e., have few nonrelevant selections) and have adequate recall (i.e., have a high percent of all available relevant items selected)? In the past, the iterative nature of the search process has helped deal with this problem, but only through continuous human interaction.

A few other simple examples of the potential need for linguistic and related skills are listed below:

- Under what conditions, in what areas, and for what purposes should particular sources or authors be given precedence over others when the computer must make a decision?

  This problem clearly requires both an understanding of alternative sources and their relative strengths and weaknesses, and an understanding of how to present this information to the PC in an effective manner.

- What patterns of words, phrases, and other parameters work best for *me* if I wish to have the computer give special weight to items that are "like" certain other ones that I have found particularly valuable? I do not want simply to duplicate the substance of the example items, but to seek some new substance presented with a structure or "feel" that has worked well for me.

  This is a longer-range possibility. We all can sense such items when we encounter them; they are what often leads to personal preferences for particular authors, editors, or publishers. However, partly because we have had no good reason so far, we have yet to develop the skills needed to instruct a computer how to recognize such material when it encounters some.

What the new skill requirements seem to mean is that, after cycling through a high-tech period when rather esoteric computer-related skills were the ones most needed, we may spiral out at a position just above where we started. One of the most critical of future user skills could be for a real depth of understanding of the capabilities and limits of language – for an expanded form of literacy in a nearly classical sense. Coupled with the need for an understanding of how one's own thought processes are best stimulated (not a bad rendition of the old injunction, "Know thyself!"), the PC suddenly shows at least the possibility of shedding its technical past to blend quite intimately with the humanistic tradition. Or, better yet, to become part of the still too-weak bridge between the "two cultures" described by C.P. Snow.[29]

[29] Snow, C.P., *The Two Cultures*, The New Statesman and Nation, 1956.

APPENDIX A

ELEMENTS FOR STRUCTURING THE PURPOSEFUL PROCESSING OF INFORMATION

1. PROCESSING SUBSTEP DESCRIPTIONS

The descriptions given below are in the same order as in **Table 2**. It should be noted in advance that, in some cases, there is overlap and a degree of merging of several closely related substeps. As a result, some situations may require judgments to be made, but the process has caused no problems in the uses of the taxonomy we have made so far. Avoidance of these overlaps would require a far finer and less convenient categorizing of substeps.

A. ACQUIRE: To come into possession of an information item in a condition ready for examination

1. **by UNCONTROLLED/PASSIVE RECEIPT** – Nonroutine, unordered items; sometimes valuable but often of the type described as junk mail.

2. **by "NORMAL"/ROUTINE RECEIPT** – For example, inquiries, orders, invoices and bills; business letters, memos, periodic and special reports, and other routed items; and similar types of materials. This category also can include items held by the user in a fully available state and, in particular, items created earlier by the user for possible future use (or reuse).

3. **by RESPONSE TO CONTINUING REQUESTS** – Deliveries of media subscriptions and library, database, and other materials selected for *continuing* delivery to a user on the basis of a user-specific interest profile (or an equivalent type of personal "needs" definition).

4. **by RESPONSE TO SPECIAL, ONE-TIME REQUESTS** – A variety of types of personally purchased, rented, or ordered materials, including specific items defined for selection (by a user) to meet a one-time (or short period) need.

B. REVIEW: To gain, through examination, enough understanding of the substance of an acquired item that the next appropriate activity for that item can be determined

1. **by MINIMAL EXAMINATION** – To establish the identity and nature of contents of an item, and to determine (or confirm) the level of user interest in its substance.

2. **by CASUAL EXAMINATION** – A "browsing" type of review for stimulation or background acquisition, or to check on activities only peripherally related to one's primary responsibilities or interests.

3. **by ACTIVE EXAMINATION** – The normal or "duty" type of check of materials closely related to one's responsibilities or interests.

4. **by SPECIAL EXAMINATION** – The detailed review of new, unusual, "exception," or especially critical types of information. Special Examinations often result from "reclassifying" Active Examinations (and occasionally Casual Checks) because of some observation made during these reviews.

C. **SEARCH and RETRIEVE**: To identify, locate and arrange for acquisition of information substance that is relevant to one's situation

1. **when RIGOROUSLY DEFINED** – For a need that is well understood, and where the required substance is easily and unambiguously indexable.

2. **when "FUZZILY" DEFINED** – For a need that may be well understood, but for which the relevant materials are not easily or unambiguously indexable.

3. **when RECOGNIZABLE-IF-FOUND** – For a need that is not easily described, but for which potentially useful materials, *if encountered*, will be easily recognized.

4. **through BROWSING** – Where searching is done primarily for substance that can provide stimulation, or broad background acquisition, or, with luck, fortuitous "discovery."

D. **OPERATE**: To take action (once or repeatedly) with the purpose of adding value to and/or deriving value from, the substance of an item

1. **involving PRIMARILY GENERAL ABSORPTION OF SUBSTANCE** – For background or stimulation or entertainment, with a level of attention that can range from relatively low to quite intense. General Absorption of background is a prerequisite for essentially all other types of Operations (and, in this role, can overlap considerably with Special and Active Reviews). However, it also is the primary, and often the only Operation for many classes of materials, such as recreational and background reading and video entertainment items.

2. **using ENTRY, CORRECTION, SELECTION, and TRANSFER PROCEDURES** – To enter and verify substance in an organized way, to edit entered or received substance, and/or to transfer some or all of the substance of an item to where it can be more useful for conducting subsequent information activities or other action steps. The transfers can involve movement of the substance into one or more other items, to a new physical location, or into a new format.

3. **using GRAPHICS/IMAGE CREATION TECHNIQUES** – To develop visual representations of information such as engineering drawings, business and scientific graphics, two- and three-dimensional

pictorials, diagrams, and lecture or presentation materials. Inputs can be in the form of raw data or information that has been processed through other operations described below.

4. **using ORDERING or RANKING or OTHER TYPES OF STRUCTURING** – To conduct qualitative examinations or analyses, either as a starting point for other Operations or as an end in themselves.

5. **using COMPUTATION or SIMULATION or OTHER ALGORITHMIC APPROACHES** – To conduct all types of quantitative analyses.

6. **using PATTERN IDENTIFICATION and RELATIONSHIP DEVELOPMENT TECHNIQUES** – To integrate materials and/or partial "solutions" to help establish findings, reach conclusions, determine implications, and prepare any appropriate recommendations.

7. **involving PATTERN CREATION** – To conduct research to build new and more useful perspectives, concepts, and approaches.

8. **involving FILING, INDEXING and STORING** – To maintain selected materials in a way that makes them accessible for later retrieval and use.

E. **COMMUNICATE:** To prepare substance for passing on to others

1. **with TEXT AND NUMBERS** – In the form of a computer entry.*

2. **with TEXT, NUMBERS, and GRAPHICS** – For controlled and restricted distribution, as is common for internal business materials.

3. **with TEXT, NUMBERS, and GRAPHICS** – For open, widespread distribution, as is the goal for most print publisher materials.

4. **by ORAL DELIVERY** – In one-to-one or one-to-few situations, such as during telephone conversations of all types, and during business interviews, small conferences, and internal reporting sessions, or while participating in social conversation.

5. **by ORAL DELIVERY** – In one-to-many situations, such as at large meetings and for broadcast performances and reports.

6. **by ORAL DELIVERY** – In high interaction conditions, such as at small, informal meetings, where strong interaction with others is an important part of the meeting objective; examples include "brainstorming" sessions and conflict resolution activities.

F. **DISCARD:** To remove an item from application of any further information activity by a particular user/recipient

---

* Note that #1, #3, and #5 of the COMMUNICATE activity are frequently performed during work that is carried out as part of #2 of the OPERATE activity.

1. **by DIVERSION** – To another specific individual or activity more appropriate for dealing with the source or substance of the item.

2. **by "OUT BASKET"** – For secretarial disposal.

3. **by TRASH BASKET** – For normal (or sometimes shredder) disposal.

4. **by NATURAL PROCESS** – Such as disappearance in real time (as in continuous control operations), or by user departure from the scene (when leaving a cinema or changing a TV channel).

2. EXAMPLES OF APPLICATION

One means to test the this categorization of subactivities is to apply them to a variety of generic real life situations and see if they offer reasonable matches to what we experience as happening. We illustrate this test in **Table A-1**, which contains a collection of examples that indicate how different types of users process information through the various steps to achieve different purposes.

Most of the examples are largely self-explanatory. However, to illustrate the interpretation procedure, a sample of detailed descriptions are listed below.

Example 1: General Manager Receiving a Telephone Call

Acquire: Receiving a phone call usually will represent a "Normal" or Routine receipt* (#2 in **Table A-1**). There will, however, also be some call-backs that result from an earlier call made by the General Manager (#4), and there may be some poorly-screened, unsolicited calls (#1) that filter through the control provided by the manager's secretary.

Review: A preliminary Review usually will have been conducted by the General Manager's secretary. This Review may lead to a Diversion of the call to another person, or a Discard action that does not allow the call to go through. As a result, the General Manager's Review is likely to be an Active one. However, sometimes the situation will merit Special attention, and there will be occasions (such as when the call is accepted primarily as a matter of courtesy) when a fairly Casual review is all that is needed.

Search and Retrieve: Normally, time does not permit any substantial Search for support materials when a call is received. Good secretaries sometimes can locate and provide background information quite rapidly, and some computer systems are beginning to be

* Note that the terms "Normal" and "Routine" refer to the appropriateness of the call, and not to the urgency of the substance that will be conveyed. The call can involve reporting on a major emergency and still fit this category.

Table A-1

Examples of User Activity Patterns

USER SITUATION	ACQUIRE	REVIEW	SEARCH/ RETRIEVE	OPERATE	COMMUNICATE
			WORK SITUATIONS		
GENERAL MANAGER • Memorandum	In-box, or provided at briefing or meeting **[2], some [3], [4]**	Mostly active, but can involve mix of others **[3], some [2], [4]**	Some requests—all made through intermediaries **[1], [2]**	Primarily absorption and pattern-related operations **[1], [6], some [4]**	Mainly oral at small meetings, also often by memoranda **[4], [5], [2]**
GENERAL MANAGER • Telephone call	Mostly routine; some unwanted or return calls **[2], some [1], [4]**	Identity and some purpose checks, usually by secretary **mainly [3], some [1], [4]**	Possible letter/file on relationship; limited by search time constraints **some [1] if practical**	Primarily absorption and pattern-related operations **[1], [6], some [4]**	Oral one-on-one (by definition), plus some record of call **[4], some [2], [5]**
STAFF ANALYST • Specific study	Typically in response to analyst request **[3] and [4]**	Careful review for content and relevance **[3] and [4]**	Searches to meet all types of needs **[1] - [4]**	Can involve all types of operations **[1] - [7]**	Briefings and reports **[2], [4], some [5]**
STAFF ANALYST • Periodic report	In-box receipt; may result from request **[2], some [3]**	Mix of casual and active reviews **[2], some [3]**	Occasional searches and profile changes **[1], some [2]**	Much absorption, but some of all types of operations **[1], [2] - [7]**	Mainly discussions and memoranda **[2] and [4]**
STAFF ANALYST • Report writing	Creation using held, pre-prepared, and new materials **[2], some [4]**	Careful check to help organize work **[3], some [2]**	For some special needs **[1]**	Some of all types of operations **[1] - [7]**	Written report, by definition **mainly [2]**

Key to Numbers Shown Above in Brackets (shortened versions of Table 1 categories)

ACQUIRE	REVIEW	SEARCH and RETRIEVE	OPERATE		COMMUNICATE	
1. Uncontrolled	1. Minimal	1. Rigorously defined	1. Absorption	5. Quantitative analysis	1. Computer entry	5. Oral: one-to-many
2. Normal/routine	2. Casual	2. "Fuzzily" defined	2. Enter, correct, etc.	6. Patterns/relationships	2. Text: small distribution	6. Oral: highly interactive
3. Continuing request	3. Active	3. Recognizable	3. Graphics/images	7. Pattern creation	3. Text: large distribution	
4. One-time request	4. Special	4. Browsing	4. Qualitative analysis	8. File, index, store	4. Oral: one-to-one/few	

Cont. ➤

Table A-1

Examples of User Activity Patterns, cont.

USER SITUATION	ACQUIRE	REVIEW	SEARCH/ RETRIEVE	OPERATE	COMMUNICATE
			WORK SITUATIONS		
RESERVATION ORDER CLERK	Telephone request [2]	Quick decision that call is appropriate [1]	Retrieve credit and availabiltiy records [1]	Primarily entry-related operations [2]	Text and numbers to computer [1]
DATA ENTRY CLERK • e.g., bill payments	In-box receipt [2]	Quick review for completeness and general accuracy [3]	Occasional look-up of associated records [1]	Primarily entry-related operations [2]	Text and numbers to computer [1]
AIR TRAFFIC CONTROLLER • En route control	Continuous data display [2]	Continuous check on need for action [3], **some** [4]	Occasional look-up of associated records and potential impacts [1]	Much absorption and pattern analysis; some quantitative and qualitative work [1], [5], **and** [3], [4]	Mainly oral via radio; some data transmis-sion [4] **and** [6]
			HOME SITUATIONS		
CONSUMER • Home VCR	Purchase/rental, often after search [4]	"Identity" check of item [1]	For selection prior to acquisition [1] - [4]	Primarily general absorption [1]	Social (if any) [4]
CONSUMER • Home newspaper or magazine	Subscription, often after search [2]	"Identity" check and/ or browsing [1], [2]	For selection prior to subscription [1] - [4]	Primarily general absorption [1], **some** 6, [8]	Social (if any) [4]

Key to Numbers Shown Above in Brackets (shortened versions of Table 1 categories)

ACQUIRE	REVIEW	SEARCH and RETRIEVE	OPERATE		COMMUNICATE	
1. Uncontrolled	1. Minimal	1. Rigorously defined	1. Absorption	5. Quantitative analysis	1. Computer entry	5. Oral: one-to-many
2. Normal/routine	2. Casual	2. "Fuzzily" defined	2. Enter, correct, etc.	6. Patterns/relationships	2. Text: small distribution	6. Oral: highly interactive
3. Continuing request	3. Active	3. Recognizable	3. Graphics/images	7. Pattern creation	3. Text: large distribution	
4. One-time request	4. Special	4. Browsing	4. Qualitative analysis	8. File, index, store	4. Oral: one-to-one/few	

programmed to offer such support. In both cases, the information is of category #1, "Rigorously Defined," in **Table A-1**.

Operate: The primary operations will be those of General Absorption (#1) and application of the General Manager's skills at identifying and working with Patterns and Relationships of various types (#6). There may be opportunities to conduct some simple qualitative analyses (such as ranking and structuring relevant considerations) during the conversation (#4).

Communicate: The conversation itself is Communication of type #4. It may result in some form of record or further Communication, either oral (also usually type #4) or written for a small group of associates (#2).

Example 2: Staff Analyst Receiving a Periodic Report

Acquire: Periodic reports usually will be received as Routine items (#2), by delivery to in-boxes or over an internal electronic mail system. In some cases, the staff analyst may receive the report because he or she submitted earlier a Continuing request (#3).

Review: Most periodic reports will tend to receive rather Casual reviews (#2), except when received by persons with direct responsibilities in the area covered. Such persons often include staff analysts, with regard to some of the periodic reports they receive; if this is the case, their reviews normally will be Active (#3).

Search and Retrieve: Searches for additional materials usually are not needed by a staff analyst when he or she is examining a periodic report. If needed, the searches usually will be for supporting detail and/or explanations of exceptions or discrepancies. They will mostly involve Rigorously Defined items (#1), but in some situations more Fuzzily Defined materials (#2) may be sought.

Operate: The primary, and in most cases the only, operation will be General Absorption (#1). If more action is taken, it can involve any or all of the types of analysis (i.e., #3-#6), and/or Storage for future reference (#8) if the item appears to have potential future value.

Communicate: Often, no communication will be involved. If there is some, it is likely to involve a memo with limited distribution (#2) or a conversation (#4).

Example 3: A Consumer Receiving a Magazine

Acquire: When a consumer receives a copy of a magazine by subscription, as opposed to buying it at a newsstand, the receipt obviously is regarded as an expected, routine event. However, because the delivery results from the earlier act of subscribing, it is better classified as a Response to a Continuing Request (#3). Review: The review will be little more than an "identity check" and perhaps a brief browse to get a sense of the contents of the specific issue. Thus, the Examination is Minimal (#1) or, at most, Casual (#2).

Search and Retrieve: Usually, there will be no Search of any significance related to the receipt of a magazine. Note that there may have been an earlier search since, at the time of subscribing, the consumer may have obtained a number of magazines to compare when deciding what to purchase, and may have "shopped" a bit to get favorable subscription terms.

Operate: The primary operation is General Absorption (#1). For hobby and relatively serious magazines, some activities may include interpretation of information (#6), and the Storage (#8) of articles or complete issues.

Communicate: If there is any communication, it will be social and conducted through conversations (#4).