CONVERSATIONAL COMPUTERS

William D. Orr
SF Associates

John Wiley & Sons, Inc. New York | London | Sydney

Library of Congress Catalog Card Number: 68-30916
GB 471 65708X
Printed in the United States of America

CONVERSATIONAL COMPUTERS

Preface

Much has already been written and said about what computers and automation are doing to our society. One of the loudest and clearest messages coming out of the community of computer experts in the mid-1960s is that even if all this writing and saying were perfectly true in every particular it now stands in danger of being swamped in the implications of a "new wave" of computing technology. One of the welcome features of the new wave, however, is that you do not have to understand the "old wave" to join in the discussion. The following parody may be useful in getting a feeling for what is happening.

Suppose that everything that has happened is approximately the same as we know it from experience, except that the history of the internal-combustion engine is as follows:

In all the preceding centuries men had to transport themselves and their burdens on foot. They rigged up sleds and slings and other aids to ease their work, but basically transportation technology remained the same. Then about 1944, as an outgrowth of wartime needs, someone invented the internal-combustion engine and hitched one to a huge trailer to help the army carry its big guns.

Engineers who saw that said at once, "We can use this kind of machine to haul our heavy laboratory equipment." This went on for several years.

About 1953 businessmen said, "If the army can use these machines to haul their big guns and engineers to haul their heavy equipment, why can't we use them to haul big carloads of products?" The first attempts were miserable failures, but after a while they got the hang of it, and later there came into being an impressive network of highways and turnpikes and service stations and mechanics. By that time many people were employed in making the trucks and trailers, for by the end of the 1950s there were several thousands of them on the roads.

Then someone said, "How can we make this wonderful new technology useful to the average man?" So they developed huge vans to carry the household goods of average men and great buses to transport hundreds of people in groups from place to place.

Then someone said, "Wouldn't it be nice if everyone had his own

means of transportation?" Much technical discussion about how to provide it ensued, and at once it became apparent that the network of highways, the service stations, and the mechanical expertise would all be useful. Design began on the first individual transportation vehicle around 1960. By 1963 the first experimental model, made largely from parts of existing trucks and vans and semitrailers, was being road-tested. In 1964 it was obvious that the individual vehicle would be a big success. In 1965 all the major truck and van manufacturers announced that they had designed a personal transportation vehicle, and in 1966 and 1967 the first private passenger cars were on the roads.

The intended implication of the parody is that personal computers will have an effect on all of us that will be at least as profound as the effect of the personal passenger car; and if we could imagine that we had been given only about 20 years in which to assimilate the internal-combustion engine and its derivatives into our way of life—instead of about 200 years —then we might be better prepared for the cultural shock we are about to experience.

Computer specialists refer to these new developments in a variety of ways, depending on their technical perspectives. Specialists in equipment hardware think in terms of "time-sharing"—meaning the apparent ability of a central computer to attend to the requests of a number of users at the same time. Programmers speak of "interactive" systems, "conversational" or "responsive" modes of usage, and "reactive" typewriters. The largest and most famous experimental system embodying the new concepts is M.I.T.'s Project MAC, which is an acronym with a double entendre: Multiple-Access Computing and Machine-Aided Cognition. Viewed from the perspective of an ultimate consumer of the whole process, the technique is sometimes called "on-line problem solving."

In the absence of consensus on a name, I have given the name "Conversational Computers" to the over-all phenomenon, which, to be sure, is only in part a matter of computers as machines; it is also part computing techniques, part programming languages and strategies, and part mindset. For the nonspecialist (actually, for the specialist, too) it is likely that his *attitude* toward the subject will pace his *understanding* of it. Accordingly, the selections in this volume have been virtually purged of technical details. As Shaw puts it in his report on The RAND Corporation's experience with JOSS in Chapter 4, "Inside knowledge is of no value to the user." If fugitive terms from the technical jargon have somehow survived in the ensuing selections they can be ignored without fear of missing an important point.

The body of thought that is brought together in this volume is in-

tended to be complete and sufficient unto its own purposes. The reader who wants to pursue a topic to greater technical depth will, I assume, turn first to the source material from which the selection of interest was extracted. There he will find the equations, diagrams, bibliographies, and other accouterments of scholarship that have been omitted here. The one exception to this editorial strategy is Silvern's "CAI in an Expanding Universe of Educational Methodology," which was written for this volume and therefore carries its own references.

The reader is advised, while reading these selections, to think of the computer as a black box somewhere in the basement, across town, or on some other continent. If the reader is handicapped by much or even a little knowledge of how computers work, he should try to forget it. Worrying about word lengths, compilers, dynamic relocation, access times, switching techniques—or even wondering where the punched cards went—will only corrupt the innocence of his approach to the subject. These caveats apply even more urgently to worries about "applications." At bottom there is really only one "application" of Conversational Computers, and that is *whatever you do during the course of a normal day.* It is as pointless to ask what Conversational Computers will be used for as it would be to ask what destinations you can drive to in an automobile or what kinds of letters you can compose on a typewriter.

Still it is allowed that readers will find some examples more congenial than others, and so we have tried in our selection process to touch on a variety of usage environments. The reader should keep in mind, however, that these are but variations on a very few basic themes: a mathematician presses a button labeled MOD and a machine computes and displays the modulus of the number specified; the button could have been labeled CPT and made to cause the computation of the cost per thousand of a specified magazine's circulation for an advertising man.

This volume is intended for the intelligent, curious nonspecialist who, in one way or another, has come to suspect that something is up in the world of computing and would like to know what—if that is possible. It *is* possible. The way chosen to "explain" Conversational Computers is to bring together in one volume some of the writings that have had pivotal effects on the field. The authors of these selections are, for the most part, specialists themselves: men respected by their colleagues for their contributions to the emergent technology. In nearly every case the selections in this volume are the same titles usually cited in the burgeoning literature by specialists presenting papers at technical conferences. How is it possible for layman and specialist alike to learn from the same tutor teaching the same lesson? The answer lies in the

historical fact that when these exceptional men prepared the material excerpted in this volume they were most often trying to share with their colleagues the excitement of new ideas which at the time transcended workaday engineering concerns. (Now these ideas have been reduced to workaday engineering concerns and are the chief occupation of a growing proportion of computer technologists. Correspondingly, the professionals' dialogue becomes more and more opaque to the layman.)

Today, however, only laymen, students, and a few open-minded specialists can have the thrill of recapturing that pristine excitement and sharing those first discoveries.

A few words are in order at this point to account for the grouping of selections under the major headings chosen: The Vision; Problem-Solving Modes; Instructional Modes; Retrieval and Query Modes; Graphical Conversation Modes; Toward the Computer Utility; and Psychological and Social Implications.

I believe that the rationale of the groupings is valid, although by no means the only way the subject could be approached. Indeed, anyone who feels strongly that the material ought to have been organized in a different way must not be surprised to find me in line as the first convert to his way of thinking.

Part One, The Vision, lays the conceptual groundwork for the ensuing selections. The excerpts by Licklider and McCarthy came from papers originally prepared for technical audiences. Both have an exciting quality of anticipation that came from the knowledge that several then-isolated developments were about to fuse in the creation of a new order of technology. Bush's article has always been an inspiration to those working in the area of mind-machine interaction, and this influence has not been dimmed by the fact that his intriguing Memex is not much closer to realization today than it was when the article was published in 1945.

The selections in Part Two, Problem-Solving Modes, deal with on-line systems in which the user manipulates the computing machinery as an intellectual tool. The goal of this kind of mind-machine interaction is to create something unspecifiable at the outset that will satisfy the individual user who is directing the process. The selections in this group describe the earliest implementation of Conversational Computers for technically sophisticated users and so may seem to be alike. Yet there are essential differences to note between The RAND Corporation's JOSS and the Culler-Fried approach. JOSS was developed because the scientists, mathematicians, and engineers at RAND (RAND!) were not using computers in their daily work so much as computer evangelists felt they would—if only special (and modestly restrictive) languages like

FORTRAN were not so much of a bother. JOSS is considered by professional programmers to be a masterpiece of elegant simplicity. Children have learned how to use it with only a few minutes' instruction, yet such is the power of Conversational Computers that after a year's experience with JOSS one senior electronics engineer wrote:

"JOSS is becoming essential to our output like paper, or coffee. It speeds up calculations; makes it easy to try experiments. It is the greatest, when it works. I have heard it compared favorably to beer and referred to as RAND's most important fringe benefit. People adjust their lives to fit around JOSS. . . . No use coming to RAND before 10:00 A.M. when JOSS arrives; in fact noon or after 5:00 P.M. is a better time, JOSS is less busy. When JOSS starts typing answers, the titillating pleasure is equaled only by the ensuing anguish when JOSS breaks off into jibberish or goes away commending your program to oblivion. We can hardly live with JOSS, but we can't live without it. We're hooked."

The Culler-Fried system, designed principally by Culler, is also a much-admired example of elegant simplicity. Yet, where JOSS deals in so-called "primitive" terms, Culler's system also places at the user's disposal the most powerful macroscopic functions he can handle. Culler's console keyboard has nine semantic levels, each of which gives a new set of meanings to each of the keys, much as the "shift" key on a typewriter changes the typed impression from a lower case to a capital letter or from "4" to "$." Culler's system also has storage places for new functions or strings of new functions invented right at the console by the user, who may be a specialist in his own discipline but know nothing about computer programming.

The proliferation of JOSS-like languages was the computer industry's highest endorsement of that approach. On the other hand, it may take years for the user community to discover the full beauty and richness of Culler's system.

In the chapter on financial analysis it can be seen that Gal's First Financial Language combines attributes of a JOSS-like language, a Culler-Fried approach to problem solving, and sophisticated information-retrieval techniques. Thus the analyst can reference an extensive collection of statistical programs, such as the growth of a stock over a 10-year period, by a single keyboard request: GROWTH (IBM.EPS. 1955–65). He can create programs at his console and store them in the system as part of his own private library. Gal's article also nicely illustrates the use of an automatic report generator capable of presenting variable data in a format created by the user to suit his personal style of analysis.

Part Three, Instructional Modes, deals with systems in which the relations between user and computer program are essentially the reverse of what they are in on-line problem solving. In such systems the *computer* is always in command of the situation, despite the fact that it may be programmed to converse with the user in beguilingly deferential language. In these systems the object is to get the user to learn something the computer already knows.

Silvern's article on Computer-Assisted Instruction (CAI) in an expanding universe of educational methodology is a rigorous though nontechnical presentation, complete enough to serve as textbook material for educators desiring to orient CAI in a context of more familiar methodology. Bolt's engaging description of computer-aided instruction illustrates one of the earliest interactive teaching programs, which is none the less remarkably fluent and flexible in its aptly named "Socratic" teaching style. HELP! illustrates how a teaching system can be embedded in a computer system's software package to enable a user to interrupt an ongoing operation to get help on a sticky point of procedure or technique. In principle, a person with no knowledge at all of, say, FORTRAN could learn it as he uses it in specific computations.

Part Four, Retrieval and Query Modes, deals with still another kind of relation between user, computer, and stored information. In this mode the object of the interaction is to find and fetch specific units of information known to be, or believed to be, or hoped to be in storage—or, had they been known to be there, they would have been wanted. Salton's masterly survey of the canny strategems and operational motifs employed to do these things is in itself a major contribution to the art. System Development Corporation's SYNTHEX, used, among others, by the Los Angeles Police Department in the identification of criminals by cross-correlating information in crime reports, reflects the characteristic thrust toward natural-language queries in which the user can blurt out his question and expect the computer to check its key terms statistically or otherwise against the information in storage. Thompson's DEACON system, like Culler's, is rich in philosophical implications. The selection included here illustrates the use of list processing, wherein the system is able to make useful deductions from relationships that are implicit in the information, though not stored in precisely the form desired. Of more profound importance is Thompson's postulation that problem solving is done in an intellectual environment of facts and hunches and partial answers. In this deeper sense "information retrieval" corresponds, in the universe of concepts, to that more familiar process by which the physical scientist queries the physical universe. With characteristic grace and insight, Thompson gives us this model of the process:

". . . Each new step became apparent only after the previous one had been successfully taken. Further, in this process of structuring his conceptual world, all the beams of the first floor are not erected before he goes on to the second floor, as in the building of a building. Rather, he places a beam out into space, goes to the end of it, and looks around. From its tip a new beam is erected. If the new one collapses, he goes back a little. But, when he is successful and the new beam fits reality, its missing understructure (which he had presumably bypassed) appears as if by magic.

. . . There was no pattern of summaries prepared for the decision maker, or systematic collection of all the details before the start of the considerations. This fluctuating vertical movement of synthesis and abstraction is very typical in the evolution of greater understanding. More and more general syntheses are built until from the broader vantage point more relevant ramification can be seen. And then, as details are added, new pathways for connections and consolidations become clear."

Graphical Conversation Modes, Part Five, covers a subject whose implications slice across all other aspects of Conversational Computers. The impact of graphical languages on our ways of thinking has not yet even begun to be felt, in my opinion. Techniques such as those described in Part Five could well be harbingers of a general turning away from the languages of words and numbers back toward the ancient languages of pictures and ideographs, which, it can be convincingly argued, are more congenial to natural human thought processes.

The route from present number-and-word languages to a true graphical language, however, is by no means clearly marked. In Part Five the chapters by science journalists Schmedel and Pfeiffer—as well as Jacks' report of General Motors' highly regarded computer-augmented design experimentation—are descriptive, not philosophical. A theory of computer-graphical language must be built on broader experience than all the users of "graphical" systems are likely to acquire for many more years. Until a theoretical foundation has been laid "practical applications" will probably be limited to finding ways of making it easier to do with the aid of computers the same kind of graphical work now being done manually by draftsmen and illustrators.

Part Six, Toward the Computer Utility, treats the logical extension of time-sharing technology and conversational usage to large-scale information networks. Adams' KEYDATA setup, the first commercial public-information utility, serves a community of business-oriented subscribers. In this respect it resembles the MAC System at Massachusetts Institute of Technology and a number of specialized subscriber networks offered

to users involved primarily in scientific computation. But we ought not to be hasty in applying the word "utility" to the phenomenon of shared computing resources. At best, the term utility is a metaphor that brings to mind some useful generic notions from experience with power, water, and communications networks. A computer "utility" is, of course, a new kind of resource sharing. It can lead to new relations between users, new kinds of cooperative research, and structural innovations such as the "distributed university" and the "distributed business firm."

The following summary of experience at M.I.T. hints at some of these differences, within which entire new doctrines of public policy are latent:

"Three years of experience with the Compatible Time-Sharing System (including Project MAC) at M.I.T. have been a revelation in many ways. In a sense, the system and its users have developed like a growing organism. Most striking is the way the users have built on one another's work and become dependent on the machine. More than half of the commands now written into the system were developed by the users rather than by the professionals charged with programming and developing the system. The users have very generally chosen to link up with one another's private files and the public files. Whereas in conventional computer installations one hardly ever makes use of a program developed by another user, because of the difficulty of exchanging programs and data, here the ease of exchange has encouraged investigators to design their programs with an eye to possible use by other people. They have acted essentially as if they were writing papers to be published in technical journals. Indeed, the analogy is not far-fetched: an editorial board representing the community of users acts as a referee to pass on all new commands that are to be introduced into the system and on all information that is to be stored in the public files.

All in all, the mass memories of our machines are becoming more and more like a community library. The users are beginning to complain about the difficulty of finding out just what the library contains and locating the items that may be of interest to them. The facility actually goes beyond a library's usual services. It already has a rudimentary mechanism whereby one person can communicate with another through a program in real time, that is, while both are using the same program at the same time. There have been cases in which a member of the faculty, sitting at a teletypewriter at home, has worked with a student stationed at a terminal on the campus. It is easy now to envision the use of the system for education or for real-time collaboration between the members of a research team. And it does not take a long stretch of the imagination to envision an entire business organization making and

executing all its major decisions with the aid of a time-shared computing system. In such a system the mass memory at all times would contain an up-to-date description of the state of the business."*

Some of the technical, economic, and public policy questions associated with a major public utility, like the present telephone and electric power networks, are addressed with careful scholarship by Flood in Chapter 17. The Federal Communications Commission's Notice of Inquiry, Chapter 18, is highly respected by computer-industry people for its sure sense of the underlying issues involved in large-scale public networks, as well as for its statesmanlike tone. It is reasonable, however, to predict that encounters between the FCC and the historically unregulated computer industry will become more and more bruising as the investigation proceeds and regulations evolve over the years.

Psychological and social implications of Conversational Computers are discussed in Part Seven by researchers oriented toward the man, rather than the machine, side of the man-machine relationship. Baran raises fundamental questions about hidden threats to privacy posed by the coming computer utilities. Kemeny, Neisser, and Lee, however, discussing the effects of Conversational Computers on the individual's articulateness in society, seem to share the view that consequences of the new technology will be generally benign.

Having offered, hopefully, a reasonable explanation of what is included in this volume, it seems appropriate to reflect briefly on what has been left out.

First, one might expect that a book purporting to treat the subject of Conversational Computers ought to have something to say about that most ancient of conversational modes: talking. Voice conversation is not discussed because the state of the art permits only the grossest of speculations now and for several years to come. The linguistic variables and ambiguities that have for so long inhibited the growth of retrieval and query systems for written information would all be present in a spoken-language system—along with horrendous new technical problems concerned with recognizing and reconstructing human speech. By comparison, so-called "voiceprint" techniques and "canned" verbal replies generated by machines are mere child's play.

Artificial intelligence is another area of computer development that might be expected to play a prominent role in man-computer conversations. It is compelling indeed to imagine holding a more or less spontaneous dialogue with a computer in the role of intellectual companion.

* R. M. Fano and F. J. Corbató, "Time-Sharing on Computers," *Scientific American*, **215**, No. 3 (September 1966).

Before this can happen, some basic philosophical questions must first be answered that may not yet have been intelligently asked. For example, when Thompson, a superb linguistics theoretician and the developer of DEACON (Chapter 12), sets out to program on a computer the semantic network that he believes underlies and corresponds to the much-researched syntactic network, he finds that no matter how many successes he has in programming the meanings of specific words, phrases, and sentences there always remains an unresolvable doubt that one man's (or one computer's) semantic network matches another's. Each man's thoughts and expressions of thoughts are inextricably bound to a private world view derived from unique personal experiences that can never be fully "translated" for any other man.

To raise the question of how computers affect human thought processes does not require the assumption of "thinking machines" or robot intellectual companions. What we believe a computer can be used for largely determines what we actually will use it for. Before the advent of Conversational Computers a user approached the computer as a spectacularly talented solver of those problems—and only those problems—that could be fully described in advance of the attempt to solve them. The key that unlocked the computer's power was the algorithm, a precise statement of the problem in the computer's own language. One of the main features of conversational "computing" is that it bypasses programmers and algorithms altogether *in the final problem-solving stage.* The person with the problem to be solved is allowed to use intuition, guesses, and hunches as freely as he wishes. Today there are perhaps one or two dozen human minds in the whole world that have had truly intimate rapport with a mechanized information process. The news that at least some of these pioneers are bringing back from the frontier is surprising—and exciting. It is this: human imagination fatigues very quickly when freed from the routine mental "housekeeping" chores associated with thinking. For 2000 years and more thoughtful men have yearned to liberate their imaginations; a handful—da Vinci, Newton, Einstein—succeeded out of sheer intellectual virtuosity. Is it now in the power of our culture to liberate a significant number of human imaginations to fulfill their potential and, in doing so, enrich the culture still more? If it is possible, could we stand it? If we think we could stand it and want to do it, can our social learning apparatus bring it about? I have formidable doubts that the existing "establishment"—cause-and-effect logic and its mechanical handmaiden, the algorithm—can be overthrown in as short a time as a generation, even if we could somehow marshal a consensus that it should be overthrown. Nevertheless I believe it inevitable that the cause-and-effect thought processes associated with a Newtonian world

view will be superseded by looser, more spontaneous styles of thinking derived from a new world view more harmonious with contemporary notions of reality. And beyond doubt, Conversational Computers are the natural allies of freestyle thinking.

Palos Verdes, California
December 1, 1967

William D. Orr

CONVERSATIONAL COMPUTERS

Contents