The National Challenge in Computer Science and Technology (1988)

Pages
77

Size
5 x 8

ISBN
0309319277

Computer Science and Technology Board; Commission on Physical Sciences, Mathematics, and Resources; National Research Council

The National Challenge in Computer Science and Technology

National Research Council (u.s.)
Computer Science and Technology Board
Commission on Physical Sciences, Mathematics, and Resources
National Research Council

NATIONAL ACADEMY PRESS
Washington, D.C. 1988

NOTICE: The project that is the subject of this report was approved by the Governing Board of the National Research Council, whose members are drawn from the councils of the National Academy of Sciences, the National Academy of Engineering, and the Institute of Medicine. The members of the committee responsible for the report were chosen for their special competences and with regard for appropriate balance.

This report has been reviewed by a group other than the authors according to procedures approved by a Report Review Committee consisting of members of the National Academy of Sciences, the National Academy of Engineering, and the Institute of Medicine.

The National Academy of Sciences is a private, nonprofit, self-perpetuating society of distinguished scholars engaged in scientific and engineering research, dedicated to the furtherance of science and technology and to their use for the general welfare. Upon the authority of the charter granted to it by the Congress in 1863, the Academy has a mandate that requires it to advise the federal government on scientific and technical matters. Dr. Frank Press is president of the National Academy of Sciences.

The National Academy of Engineering was established in 1964, under the charter of the National Academy of Sciences, as a parallel organisation of outstanding engineers. It is autonomous in its administration and in the selection of its members, sharing with the National Academy of Sciences the responsibility for advising the federal government. The National Academy of Engineering also sponsors engineering programs aimed at meeting national needs, encourages education and research, and recognises the superior achievements of engineers. Dr. Robert M. White is president of the National Academy of Engineering.

The Institute of Medicine was established in 1970 by the National Academy of Sciences to secure the services of eminent members of appropriate professions in the examination of policy matters pertaining to the health of the public. The Institute acts under the responsibility given to the National Academy of Sciences by its congressional charter to be an adviser to the federal government and upon its own initiative, to identify issues of medical care, research, and education. Dr. Samuel O. Thier is president of the Institute of Medicine.

The National Research Council was established by the National Academy of Sciences in 1916 to associate the broad community of science and technology with the Academy's purposes of furthering knowledge and of advising the federal government. Functioning in accordance with general policies determined by the Academy, the Council has become the principal operating agency of both the National Academy of Sciences and the National Academy of Engineering in providing services to the government, the public, and the scientific and engineering communities. The Council is administered jointly by both Academies and the Institute of Medicine. Dr. Frank Press and Dr. Robert M. White are chairman and vice-chairman, respectively, of the National Research Council.

Support for this project was provided by the Defense Advanced Research Projects Agency (grant no. N00014-87-G-0082), the National Science Foundation (grant no. CCR-8619362), the National Aeronautics and Space Administration (grant no. DCR-8619362), the Department of Energy (grant no. DE-FG-05-87ER25029), and the Office of Naval Research (grant no. N00014-87-G-0110). Additional funding was received from IBM, Hewlett Packard, Cray, and Digital Equipment Corporation.

Cover: "Two Men on Edge" by Harold Cohen (1988). Photograph by Becky Cohen (1988). Painting from a drawing generated by the artist's artificial intelligence computer program, AARON. 90" × 118". Collection of Joseph F. Traub and Pamela McCorduck.

Available from
Computer Science and Technology Board
2101 Constitution Avenue
Washington, D.C. 20418

Printed in the United States of America

COMPUTER SCIENCE AND TECHNOLOGY BOARD

COMMISSION ON PHYSICAL SCIENCES, MATHEMATICS, AND RESOURCES

NORMAN HACKERMAN, Robert A. Welch Foundation, *Chairman*
GEORGE F. CARRIER, Harvard University
DEAN E. EASTMAN, IBM, T.J. Watson Research Center
MARYE ANNE FOX, University of Texas
GERHART FRIEDLANDER, Brookhaven National Laboratory
LAWRENCE W. FUNKHOUSER, Chevron Corporation (retired)
PHILLIP A. GRIFFITHS, Duke University
J. ROSS MACDONALD, University of North Carolina, Chapel Hill
CHARLES J. MANKIN, Oklahoma Geological Survey
PERRY L. MCCARTY, Stanford University
JACK E. OLIVER, Cornell University
JEREMIAH P. OSTRIKER, Princeton University Observatory
WILLIAM D. PHILLIPS, Mallinckrodt, Inc.
DENIS J. PRAGER, MacArthur Foundation
DAVID M. RAUP, University of Chicago
RICHARD J. REED, University of Washington
ROBERT E. SIEVERS, University of Colorado
LARRY L. SMARR, National Center for Supercomputing Applications
EDWARD C. STONE, JR., California Institute of Technology
KARL K. TUREKIAN, Yale University
GEORGE W. WETHERILL, Carnegie Institution of Washington
IRVING WLADAWSKY-BERGER, IBM Data Systems Division

RAPHAEL G. KASPER, Executive Director
LAWRENCE E. MCCRAY, Associate Executive Director

Preface

The Computer Science and Technology Board of the National Research Council was established in mid-1986 to identify and analyze issues associated with developing, producing, and using computers. This report is the first to be issued by the board itself: it summarizes the initial deliberations of the board and serves as a platform from which the board can now begin to launch projects with a narrower focus in such areas as computer networking, high-performance computing, computer security, software, education, and the competitiveness of the U.S. computer sector.

The report in hand combines a description of the most promising technological thrusts in the field of computer science and technology with a statement of concern about the health of the field and a call for greater and more effective implementation of computer networking. A major investment in infrastructure is needed to enhance the nation's productivity and competitiveness across all fields. The report's description of computer science and engineering highlights the significance of technological innovations made possible by computer science in the recent past and identifies promising future directions and potential obstacles.

The report is aimed at people in government, industry, and academia who are concerned about the future of computing technology as a critical area of national strength, particularly at a time when America's position in other areas is in apparent decline. The report is addressed particularly to members of the policymaking community, as they consider decisions that will influence the growth of the field of computer science and technology and future applications of computers and communications.

The report owes its existence to the devoted persistence of board member Michael L. Dertouzos, who first gave the report shape and then, on the basis of group discussions and raw material from the other members of the board, shepherded it through the numerous drafts and additional discussions that preceded its final form.

Joseph F. Traub
Chairman, Computer Science
and Technology Board

Contents

PART I THE CHALLENGE

Part I

The Challenge

1
Introduction and Summary

As the world leader in computer science and technology, the United States has been at the forefront of the evolution of this new discipline into an important, pervasive, and indispensable field that accounts for approximately 10 percent of the gross national product (GNP) and is a key factor in the national defense (CBEMA 1987). Looking forward, we see immense and growing technological promise as well as a unique opportunity to use U.S. strength in this area to enhance productivity and competitiveness across our economy. At the same time, we see the U.S. position in this field threatened from without by external competition and from within by underappreciation of the need for basic research. The challenge before us is to assure continued U.S. preeminence in computer science and technology.

We attribute U.S. leadership to a unique innovation engine. Universities and industrial research laboratories perform the basic research that fuels the entire engine; venture companies function as effective agents of early technology transfer; and mature companies develop and market the innovations that achieve widespread acceptance. In the past, government funding has played a key role in lubricating the engine. Another important ingredient of success has been the rich governmental and industrial research infrastructure consisting of advanced computational research tools.

The use of computers can boost national productivity and competitiveness across the entire economic front. This potential is evident from existing and anticipated applications. For example, in design, computers offer ever-improving simulation of industrial products to provide more knowledge of their qualities and performance prior to production, thereby improving product quality and reliability while reducing manufacturing costs. In the office, distributed systems can save time on projects involving dispersed personnel and can help to soften interdepartmental rigidities by making communication and cooperation easier. Already, computer networking shows promise for expediting the ordering and scheduling delivery of goods and services (minimizing inventory costs and production delays), the settlement of invoices, and other interorganizational transactions. Expert systems can become the power tools of tomorrow, helping professionals in engineering, finance, medicine, manufacturing, and many other fields to be more knowledgeable and productive. Robots and other computerized systems are already improving precision, production times, and consistency in manufacturing, and they will eventually allow specialized goods and services to be made at or near mass production costs.

Computers can also contribute to the increase of scientific productivity and improvements in the conduct of research. Computers are ideal tools for scientific modeling: for example, biochemists use computers to model molecular structure and behavior, gaining insights unobtainable through such earlier analytical techniques as X-ray diffraction. In aerodynamics, computers simulating wind tunnels make possible more experiments at lower cost than would be possible using actual wind tunnels. In meteorology, computers are being used to model turbulence, global atmospheric movements, and cloud behavior. And in the social sciences, computers make possible large econometric models for economists and stimulus-presentation experiments for cognitive researchers. Finally, supercomputers have so greatly transformed the conduct of scientific work that many physicists and chemists speak of computational science as an intellectual revolution equal in impact to the observational paradigm of Galileo and the theoretical insight of Newton. Computer networking has already contributed to progress in many scientific disciplines and is expected to be even more vital to researchers as services and access are enhanced.

During the next decade, with adequate investments in research we can expect that computer science and technology will make major advances in three areas in particular. The first area is that of machines, systems, and software: new developments in multiprocessors will harness many computers to a single task and will allow computers to be used more powerfully and cost-effectively across new and existing applications; developments in distributed systems will involve intercommunication of computers through local and long-haul networks; and developments in software will enhance the capabilities of existing and new computer systems. The second area is that of artificial intelligence and knowledge-based systems: progress is expected in sensory computing, i.e., machine understanding of speech and visual images; in expert systems, which represent and use expert human knowledge in specialized professional domains; in deeper cognitive systems, e.g., machines that can plan, reason, and learn from practice; and in robotics, i.e., intelligent machines that can interact purposefully with the physical world. The third area of anticipated advances is that of theoretical computer science: progress is expected in understanding the laws that govern complex computational phenomena and the limits on what is possible; this fundamental understanding will lead to the development of important algorithms and representations. These advances, together with others, will make computers more useful and easier to use.

The more sophisticated, versatile, and easy to use computers become, the greater will be their potential benefits, while the cost of a given level of performance should continue to fall. Nevertheless, the United States has had uneven success in applying computers, and it is only beginning to come to grips with social and economic changes that may accompany their growing use.

The board has developed two broad, strategically oriented recommendations on the basis of its deliberations to date, recommendations that will guide much of its work to come:

1. Enhanced, nationwide computer networking should be seen as essential to maximizing the benefits in productivity and competitiveness that are created by computers. Networking will facilitate the application and delivery of diverse advances in computer science and technology to the benefit of all segments of society. The board envisions an enhanced national information networking capability, and it has already begun to examine a host of related questions about how physically to improve data networking infrastructure; associated costs, impacts, and benefits; and the roles of industry, government, and other interested parties.

2. Investment in people to do research, identification and funding of selected grand challenges, strengthening of the research environment, and funding for basic research projects—especially in the areas of theoretical computer science, software productivity, and commercial applications of computer technology and infrastructure—should be seen as essential if the United States is to continue to lead the world in this field and to realize its promised benefits in a timely manner. Support for such basic research is currently increasingly uncertain.

The Computer Science and Technology Board proceeded independently in developing these recommendations, which are in harmony with those of the Office of Science and Technology Policy (1987) set forth in a recent report to Congress. The OSTP report called for a broad initiative to further the development and use of computer science and technology and specifically encouraged networking computers in the nation's scientific community. The board will now move to develop more specific recommendations through individual projects of narrower focus.

2

The Promise of Technology

In the mere half-century since their invention, computers have evolved from experimental curiosities to tools so widely used that today, the computer sector accounts for about 10 percent of the U.S. gross national product, and almost 10 percent of the nation's capital investment (CBEMA 1987). Table 1 indicates the size and growth of the computer equipment industry, the core of the computer sector. Initially employed only for scientific and engineering calculations and later for certain business data processing calculations, computers are now used for innumerable practical applications of numerical and symbolic information processing in areas as diverse as manufacturing, education, communications, agriculture, medicine, and defense. In the world of business, machines that were once confined to payroll and accounting are now relied upon to help create documents, route messages, analyze financial data, conduct banking transactions, handle airline reservations, run the telephone system, and gain access to the vast amounts of information stored in electronic databases. In the world of scientific calculation, machines once desired for calculating numerical tables are now used to design transportation vehicles, guide satellites, predict the weather, explore for oil, increase food production, develop new pharmaceuticals, investigate the atom, and map the human genome. In the public sector, these machines have

TABLE 1 Trends and Forecasts: Electronic Computing Equipment (SIC 3573) (in millions of dollars except as noted)

									Percent Change				
									Compound Annual		Annual		
Item	1972	1976	1980	1984	1985	1986[1]	1987[2]	1988[3]	1972–85	1980–85	1985–86	1986–87	1987–88
Industry Data													
Value of shipments[4]	6,471	10,388	26,594	53.524	55.315	53.244	57.504	63.254	17.9	15.8	-3.7	8.0	10.0
Total employment (000)	145	166	305	374	356	329	316	332	7.2	3.1	-7.6	-4.0	5.1
Production workers (000)	64.7	71.3	135	158	133	115	105	105	5.7	-0.3	-13.5	-8.7	4.8
Average hourly earnings ($)	4.19	4.91	6.98	9.77	10.40	11.20	12.10	--	7.2	8.3	7.7	8.0	--
Product Data													
Value of shipments[5]	6,108	10.136	25.658	49.275	49.998	48.848	47.857	52.642	17.6	14.3	-2.3	-2.0	10.0
Trade Data													
Value of imports (ITA)[6]	--	--	1.179	7.834	8.285	11.128	13.977	18.170	--	47.7	34.3	25.6	30.0
Value of exports (ITA)[7]	1,341	2.588	7.606	13.511	13.964	14.670	17.443	20.930	19.7	12.9	5.1	18.9	20.0
Exports/shipments ratio	0.219	0.255	0.296	0.270	0.237	0.300	0.331	0.361	0.6	4.3	26.6	10.3	9.1

[1] Estimated except for exports and imports.
[2] Estimated.
[3] Forecast.
[4] Value of all products and services sold by the Electronic Computing Equipment industry.
[5] Value of products classified in the Electronic Computing Equipment industry produced by all industries.
[6] Import data, developed by the chapter author, are on a C.I.F. valuation basis.
[7] Export data are developed by the chapter author.

SOURCE: U.S. Department of Commerce: Bureau of the Census, Bureau of Economic Analysis, International Trade Administration (ITA). Estimates and forecasts by ITA.

SOURCE: 1988 U.S. Industrial Outlook, U.S. Department of Commerce.

played a major role, not only in national defense, but also in the analysis and management of the large amounts of information involved in such government programs as the decennial census and social security. In short, they have brought about a revolution in the way we live in and think about the world and, in doing so, have become indispensable.

A series of technological innovations rapidly changed computers and the way they are used. Time sharing, which distributed computer power from a single machine in a round-robin fashion among dozens of users in numerous locations, became commercially viable in the 1960s. In the 1970s very large scale integrated (VLSI) circuits made possible the processor on a chip, which in turn made computers ubiquitous, faster, cheaper, and more powerful while computer memories grew bigger, cheaper, and more reliable. In the 1980s, the personal computer has delivered cheaper computational and storage resources directly to the end user and captivated millions of people through easy-to-learn programs for spreadsheets, word processing, data bases, and business graphics. At the same time, computer networks became more widespread, interconnecting many personal and time-shared machines, thereby redefining the computing base together with a large array of sophisticated software. As this 30-year period draws to an end, processors have become thousands of times faster at constant cost, or thousands of times smaller and cheaper at constant performance, than when the period began. Current technological trends and pioneering research activities suggest continuation, if not acceleration, of technological growth over the next decade accompanied by even more useful and powerful applications.

The purpose of this chapter is to summarize briefly the technological areas that, in the board's judgment, hold the greatest promise for influencing the field in the next decade and to speculate on the changes they may bring about in our world. The technologies identified and their potential users are discussed in more detail in Part II, along with some of the associated problems and prospects. In reading this chapter, the reader should keep in mind that we have selected key areas rather than attempting to provide a taxonomy of the field, and that we have excluded related technologies of communications, semiconductors, packaging, and manufacturing, which are also necessary to meet the challenge facing the nation in computer science and technology.

MACHINES, SYSTEMS, AND SOFTWARE

Perhaps the greatest promise lies in the evolution of multiprocessors—systems that may harness hundreds, thousands, or potentially even millions of computers to work together on a single application (for example, transcription of human speech into text). Development of this technology is motivated by the fact that current computing power is wholly inadequate, by orders of magnitude, to perform most of the interesting applications of the future. The expectations from this technology are roughly analogous to those of harnessing several horses to a cart: they are more economical than one powerful horse; their number can be adjusted to match the load; and working together they can exceed by far the power of even the strongest animal, thereby making possible qualitatively different achievements. As with horses, effective ways must be developed to harness these machines in order to exploit their power.

Multiprocessors offer generic and broad potential utility. The technology involves new computational engines that are economical, scalable, and of potentially far greater power than those available today. That power should help multiprocessors achieve ambitious new applications of artificial intelligence, such as real-time speech understanding, machine vision, learning, natural language understanding, and better machine reasoning. If successful, these applications would open an entirely new world of computer uses. Finally, by linking together a large number of the most powerful superprocessors, multiprocessor architectures might even lead to ultracomputers that could truly expand the capabilities of physical science through computer simulation of immense problems, thereby creating a new set of scientific tools, such as major computational observatories, computational microscopes, computational chemical or biochemical reactors, and computational wind tunnels.

Beyond multiprocessors, another major direction in the systems area involves the interconnection via networks of geographically remote computers into distributed systems. A distributed system can consist of a handful of interconnected machines in a modest-sized office, of a few thousand machines in a large corporation, or even a few million computers belonging to individuals and organizations throughout the country. Unlike the processors in a multiprocessor, which work on the same task under central control, the computers of a distributed system work mainly on different tasks under the control of their different users. The power of a distributed system of computers versus that of an equal number of independent noninteracting computers comes from its ability to intercommunicate in order to exchange the information needed for or supplied by the local computations. Distributed systems mirror human organizations and individuals, which, though largely autonomous in their work, communicate occasionally with one another toward achieving individual as well as common goals. Such systems therefore promise to enhance information-related functions within and between organizations, from routine office tasks to commercial transactions.

Software is needed to realize and expand computer capabilities so that they are conceptually as well as physically accessible to end users. Software consists of the collection of computer instructions that specializes general purpose computers to their applications. Programming, the task of generating software, is difficult and expensive for a variety of reasons that we discuss in Chapter 6. The expected proliferation of multiprocessors and distributed systems will create further software and programming challenges. Nevertheless, development of software may achieve greater progress than in the past through development and use of tools and techniques for increasing software productivity, a crucial goal for researchers and industry.

Reaping the benefits of computers will depend on improving the interface with the user. Graphics and visualization are one source of the necessary improvements. Until recently, computer graphics had been used primarily by scientists and engineers. With the large-scale introduction of personal computers and workstations with bit-map displays, graphics is fulfilling its promise. Direct manipulation of objects on the screen is replacing traditional, much less user-friendly interaction via typed command languages. As a result, sophisticated computer technology is becoming widely accessible to casual users, lay persons, and even young children.* Improvements in graphics, visualization, and the user interface in general will draw on advances in computer hardware and software and on inputs from cognitive psychologists and other experts on interactions between people and machines.

Graphics and visualization are having a particularly strong impact on scientific research, especially when supercomputers, which

---

*Much of this technology, popularly associated with the Apple Macintosh, emanates from developments at Xerox Palo Alto Research Center beginning in the early 1970s.

generate and process massive amounts of data, are involved (McCormick et al. 1987). Fields such as molecular modeling and computational chemistry, solids modeling for mechanical engineering, computational fluids dynamics, and computational astronomy require visualizations of considerable complexity involving the use of color-shaded, (pseudo-) realistically portrayed objects and data. Advanced applications are beginning to call for animated as well as static images.

ARTIFICIAL INTELLIGENCE

The expected growth of multiprocessors and continued research strides in speech understanding and machine vision are expected to advance sensory computing substantially during the next decade. To the extent that sensory systems become as successful as we expect, they will have a dramatic impact on the way people interact with computers, since speech and vision, unlike typing, are natural means of human communication. Thus, sensory systems can make computers easier and faster to use and therefore accessible to a wider range of people than they are today.

Another important aspect of artificial intelligence (AI) is expert systems. These systems represent and use human knowledge for the solution of problems in specialized domains that are difficult enough to require significant human expertise for their solution. For example: in manufacturing, a well-known expert system is given a customer order for a computer installation and then designs a manufacturable configuration of the subsystems and schedules production; in finance, expert systems assist bank officers in deciding the credit worthiness of loan applications. Currently available expert systems present only the beginnings of what may someday be possible. The evolution of more powerful multiprocessors, paradigms, and algorithms supporting ongoing research in the representation, acquisition, and utilization of the knowledge needed by expert systems is expected to increase their usefulness and ubiquity, blending them into the general stream of computer systems and applications.

Machine intelligence is also advancing through deeper cognitive systems and, in particular, machine learning. Unlike expert systems, which are preprogrammed with expert knowledge, learning systems are capable of learning from the tasks they perform through practice, much as people do.

Finally, robotics is another technology that involves machines that are sufficiently intelligent to interact with the physical world to perform designated tasks. Robotics builds on sensory computing as well as mechanisms capable of subtle motions. The evolution of effective robotic systems could result in several benefits—increased factory productivity and, perhaps as significantly, the ability to produce individually tailored products at mass production costs.

THEORETICAL COMPUTER SCIENCE

As a young discipline, computer science is in the process of building up its theoretical base and will probably continue to do so for many years to come. Until that base of theory is more fully developed, we will be able to use computers to solve only a tiny fraction of known problems in theory and applications. The utility of theory in computer science, as in other more mature fields, is that it helps to order and explain complex phenomena through simple laws, it discovers limits on what is possible, and it guides the discovery of new principles and new possibilities for computers.

To date, there have been important advances in the areas of computational complexity, which considers the intrinsic difficulty of solving a given problem; in algorithms and their analysis, whereby new procedures are sought to solve difficult problems; and in semantics and languages, whereby as a result of theoretical insights, certain important system programs (compilers) are now routinely constructed. Theory has also contributed to cryptology, whereby methods have been developed for ensuring the privacy and authentication of computer messages.

CONCLUSION

The technological developments highlighted above, along with the existing technological base of computers, paint a picture of formidable prospective tools for the Information Age. However, if these tools are to be widely available and truly effective, several steps must be taken to facilitate their development and use. Basic research and better, more widespread data networking are two of the most important such steps.

3

The Promise of Infrastructure

The board believes that the effective use of computer technology will increasingly require a networking infrastructure. We envision a nationwide computer communications capability that would enable any computer in the United States to communicate with any other computer easily, reliably, and over a broad range of speeds commensurate with individual application needs. This capability could accelerate the conversion of computer technology advances to practical uses in businesses and homes, and it could help businesses and other organizations increase their productivity.

The board has addressed some of these issues in reviewing federal proposals to improve networking for U.S. researchers (CSTB 1988). It sees in that narrow context powerful options for improving the way the community at large does its business. In advance of an effort to study the issues associated with that much larger goal, however, we review the motivation for our interest in improving networking on a national scale.

INFORMATION NETWORKS

Almost 50 years elapsed between the invention of the telephone in 1876 and completion of the national toll network. The first nationwide plan for providing good random access service was put forth in 1925. A long time, perhaps, but today we take voice telephone service for granted. We can call anyone, anywhere. We know how to operate virtually any telephone anywhere. If we do not know a person's number, we can look it up in a telephone book or call directory assistance. Once a call is connected, we can talk as fast as we are able, and we can use any language we wish. The telephone network handles it all with ease. Our commerce, our pleasures, and our everyday life have come to depend on the existence of this richly connected, ubiquitously available voice communication highway system.

By contrast, almost none of the characteristics just cited applies to the communication of information between computers. The computer networks of today, and often individual computers, are islands unto themselves. They are not interconnected, and it is not possible to send information between any two systems. The members of the Computer Science and Technology Board, for example, are members of leading institutions in computer technology in our nation, yet we are often unable to send electronic mail among ourselves without great difficulty and without using pathways and circumventions unavailable to the general public.

To understand this, consider that, in order to make a voice telephone connection, in principle it is necessary only to connect two electrical wires. To allow computers to communicate is much more difficult because, in addition to physical connectivity, there must also be logical connectivity. To explain: computers transmit and receive blocks of bits, which must be packaged in particular formats. For example, each package must contain the address of the intended recipient, the sender's or the return address, the data being communicated, so-called check bits intended for error control (one erroneous bit can ruin a message with millions of bits), as well as other routing and control information. Designers of communicating systems must agree on exactly how these packages of bits are to be constructed. In addition, every user must agree on the sequence of steps required to establish a connection, and on what procedures must be followed in the event of a transmission error or under any of a number of other exceptional conditions.

The collection of formats for packaging data and the rules that govern the logical flow of data transmission are called data communication protocols. While there has been a significant international effort to standardize on several protocols, the attendant development of consensus has taken too long relative to the pace of computer technology, and many small networks have been constructed using many different protocols. In effect, these different networks do not speak the same language and are unable to communicate with one another, except perhaps with a great deal of effort.

In computer networking, just as in transportation networks, the speed at which individual users can operate is an important consideration. For decades the nation lived with and became accustomed to a teletype network that transported data at about 150 bits per second, a rate roughly equivalent to fast human typing. In the 1960s, computer modems were introduced to connect computers to the voice telephone network. Starting at rates of 110 bits per second, these devices have evolved so that today most data traffic is transmitted over voice facilities at 1,200 bits per second. At this speed, a page of text appears on a display screen in about 8 seconds. While this speed may be appropriate for displaying messages, there are many applications, especially in computer-to-computer communication, which would thrive on greater, even much greater, speeds.* Imagine trying to skim through a book to find a particular section if it took 8 seconds to turn every page or trying to access remote supercomputers that consume and generate data at millions of bits per second.

The extensive wiring of the nation with optical fibers by the common carriers represents an important national facility for high-speed computer communication. The latest fiber systems have data rates of 1.7 billion bits per second, equivalent to some 50,000 simultaneous voice telephone calls, on each hair-thin fiber. Moreover, the progress in lightwave transmission during the last decade has been such that the transmission capability of fibers has doubled each year. We expect rapid progress to continue for at least another 5 to 10 years. We also welcome the growth in digital communications services and, in particular, the movement toward voice-data integration, network standards such as the Open Systems Interconnection (OSI) model, and higher speed services available to individuals (e.g., through integrated services digital networks (ISDN) being introduced gradually by common carriers). But the current and anticipated situation for computers is very much as if we had built a superhighway system spanning the nation without on and off ramps, without a connecting

---

*A high-resolution screen in today's workstations often displays a mixture of text and graphics comprising several million bits of information. A typical program used even for such mundane purposes as word processing is of roughly the same size, measured in bits. Engineering drawings and photographic images are yet other examples of information requiring millions of bits to describe. We cannot transmit quickly or economically these kinds of information over current long-distance data networks.

network of local access roads, and without common understanding of vehicle speeds, widths, and loads.

A national information networking capability could build on evolving digital communication facilities. It could extend access to a number of information and network-based services now only available to relatively affluent computer users and make possible new public-access services that can only be sustained if done on a sufficiently large scale or that require state-of-the-art networking technologies (see Figure 1). If the history of the highway and telephone systems is any indication of what the future may hold, a new infrastructure based on computer networking might also result in an array of new information industries and businesses that we cannot even foresee at this time. Nevertheless, in what follows, we describe some of the uses we envision from our current vantage point.

USES OF INFORMATION NETWORKS ON A NATIONAL SCALE

The potential uses of expanded, nationwide computer networking are suggested by existing systems as well as by our understanding of emerging technologies. Information and information-related services are already bought, sold, or otherwise transacted within and between enterprises. For example, many traditional service industries (e.g., finance, insurance, accounting, and law) depend heavily on information as a product or a component, while newer services (e.g., electronic mail, electronic access to bibliographic data bases, and large-scale systems design and integration) have emerged as a result of developments in computer technology and its uses.

Existing networks also provide glimpses of the potential of networking to enhance productivity in the manufacturing sector. Computer networks can facilitate collaboration among dispersed design teams and enhance interaction among distributed design, manufacturing, and marketing personnel. Such activities have begun to emerge at the most progressive companies, and we expect their effectiveness to grow with expert systems and multiprocessors. Commerce can be made more efficient by the electronic exchange of orders, invoices, and payments, in lieu of much slower exchanges by conventional mail. The automation of ordering, invoicing, and so on, is called electronic data interchange (EDI). Today, some firms use their own networks for EDI, while others rely on shared networks and

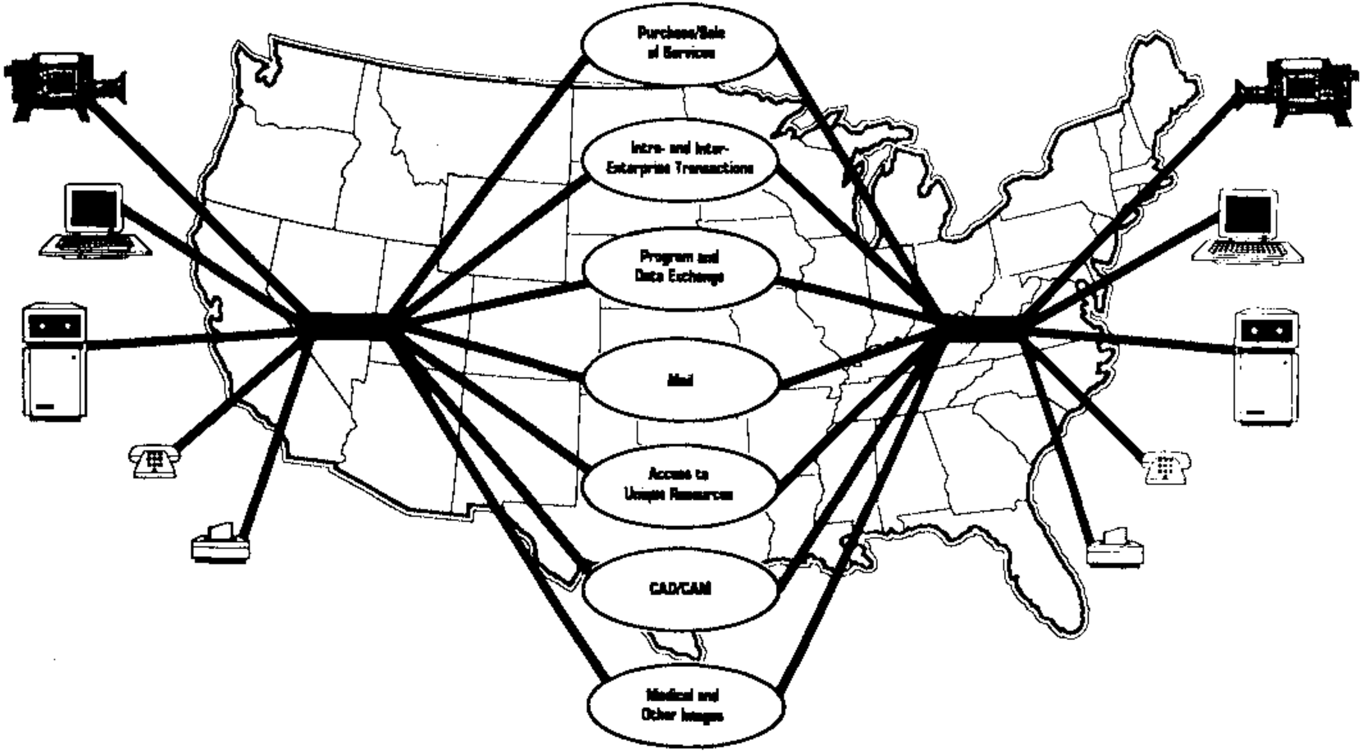


FIGURE 1 Enhanced, nationwide computer networking could make a growing variety of information services available to all segments of the economy.
Provided through the courtesy of Pacific Bell (a Pacific Telesis company). Alan Marcos, artist.

third-party computer services. In either case, the computer networks they use for such transactions are limited in scope and service.

Networking on a national scale would support (economically) the establishment of a variety of computer-based, public access information resources, such as a national digital library which might make it possible for anyone, anywhere in the nation, to access and read any book, report, magazine, or newspaper (at a nominal cost). In the research community, we have already seen how achieving the benefits of supercomputers is closely intertwined with networking, which allows remote access to these scarce and costly facilities as well as collaboration among dispersed researchers (CSTB 1988). Realizing the potential for new, information resources would require not only advances in networking, but also advances in storage, retrieval, coding and classification systems (to facilitate information access and maximize its usefulness), and user interfaces.

Other countries, smaller in scale and slower to deregulate their communications industries, have established more or less nationwide public data networks run by governmental postal, telephone, and telegraph authorities (PTTs). Those networks have offered limited protocol, equipment, and speed support, but in some cases PTTs have begun to explore the potential for nationwide information services. Perhaps the most frequently-cited example is the videotext service in France that combines Minitel terminals (provided by the government) with a public data network. Through Minitels, many information-based services are available through a uniform mode of access to consumers: telephone directories, travel and entertainment schedules, shopping, and more, but only a few have large customer followings. The Minitel experience underscores the challenges of implementing information services on the large scale and in the free-enterprise environment that characterize the United States, let alone for providing the richer service offerings that emerging technologies can make possible.

In this country, the greatest progress toward nationwide, inter-organizational networking service has been achieved in the research community. For example, the government launched Arpanet through DARPA in 1969. It has been a model for subsequent public data networks, including those offered commercially; and in the research world, it has been complemented and augmented by such general-purpose networks as Bitnet and multiple special-purpose networks (e.g., the Space Physics Analysis Network). The fragmentation of research networking, the low quality of many research networks, and the benefits of nationwide interconnectivity for researchers, have led to proposals for a national research network with widespread accessibility, high speeds, and a variety of associated information services (OSTP 1987). But even this specialized project raises many questions about costs, financing, the role of computer, communications, and other companies, the role of the government, and so on, as well as appropriate technology (CSTB 1988). Note that the goals and benefits of a national research network can be met by a loose federation of smaller, private and public networks; "national" should not necessarily be taken to mean monolithic or even government-owned.

Serving the economy as a whole, public data networking (through value-added networks) and such network-based services as EDI and electronic mail are commercially available in this country. The development of the markets for these services has been slower then projected and, for much of this decade, unprofitable. The growth of these markets is tied to growing recognition of the benefits of networking computers, growing comfort among a wide range of people with the use of computers, as well as the spread of computer equipment and the development of applications that combine data processing with communications. It is also expected that emerging standards (e.g., standards for EDI document formats, for voice-data integration, for logical connectivity of people engaged in applications, and for electronic mail system interoperability) will contribute to the demand for network-based services.

The board recognizes that achieving its vision of nationwide computer networking with greater speed, logical connectivity, and accessibility as well as a richer menu of services than now available raises many questions including the following:

- What are the principal technical obstacles, and what would be involved in overcoming them?
- Does nationwide service require a single physical network?
- Is industry likely to supply the necessary features and services on its own, and if so, in what time frame?
- What would this national capability cost, and how should it be paid for?
- What social, economic, and legal side-effects and adjustments might be involved?
- How can networking be made easy to use for all prospective users without compromising the privacy and security of users and their applications?

- What would be the most effective roles of government, industry, and other entities?

None of the above problems is insurmountable, but, as the above list suggests, planning and management as well as technology will be important in enhancing computer networking on a national scale.

4

A Unique Innovation Engine

HOW AND WHY THE ENGINE HAS WORKED WELL

The technologies and infrastructure discussed in the preceding sections are essential tools for meeting the national challenge in computer science and technology. They are based on the unparalleled record of achievement and innovation in U.S. computer science and technology during the past three decades. If we are to face the challenge before us, we must continue that record. To do so we must maintain the innovation engine that we have built and operated so successfully.

In simple terms, what we call the U.S. innovation engine consists of three components: universities, venture companies, and mature companies. The federal government—primarily through research and development funding from DARPA and NSF and to a lesser extent from the Office of Naval Research (ONR), the Air Force Office of Scientific Research (AFOSR), the Department of Energy (DOE), the National Institutes of Health (NIH), and the National Aeronautics and Space Administration (NASA)—plays an essential role in lubricating and tuning the engine and in determining its long-term future. Government has a direct interest in computer science and technology as a customer and as a user. Moreover, by influencing the health of the overall economy, government affects the environment for privately funded research and development as well as private-sector market development. Finally, government has also advanced computer technology by underwriting large projects that link companies and universities to produce novel systems and prototypes.

The universities perform most of the basic research that fuels the engine, and they supply the talent required to pursue research and development activities across the economy. Because they are organized specifically to seek new knowledge and disseminate it, universities are where novel and innovative ideas have been and are most likely to be hatched. Over the last three decades, the Defense Advanced Research Projects Agency has played a leading role in funding basic research in computer science at universities. DARPA's strategy has stressed the funding of a few high-risk visionary projects and the building of a critical mass of proven research talent at relatively few locations. And through most of its history, DARPA has invested in technologies that have proved to have both military and civilian applications, thereby using research dollars to stimulate commercial development. DARPA's approach has produced landmark innovations that include time-sharing, artificial intelligence and expert systems, computer graphics, VLSI design tools, packet-switched networks, and, more recently, new architectures for multiprocessor and distributed systems. NSF has also been a major contributor to university-based basic research and is credited with funding several important advances in theoretical computer science as well as supporting experimental computer science and upgraded educational facilities. Smaller yet significant contributions to basic research have been made under funds provided by the military services, NASA, NIH, and the Department of Energy. These innovations represent some of the most important thrusts in computer science and technology over the past 30 years, and they account for the successful operation of the university component of the U.S. innovation engine.

Successful world leadership in computer science and technology would not have been possible without venture and mature companies. Landmark industrial innovations include modern semiconductor technology, the microprocessor, the personal computer, rotating and solid state memories, supercomputer architectures, as well as several materials, packaging, and manufacturing breakthroughs.

For most companies, the pressure to sustain and increase profits makes it difficult to justify long-term research. Moreover, companies are loathe to invest in acquiring new knowledge that might accrue to the benefit of outsiders, particularly competitors. This is especially the case with the fast-paced computer technology, where the head start advantage of the innovator may be short-lived. Venture companies, which are often spin-offs from university computer science departments or from large companies, translate the results of research into leading-edge products and get technological innovations into the marketplace relatively quickly. Venture companies do virtually no research, lacking the time or the resources to do anything other than get their new products to market. Mature companies, which in many cases began as smaller venture companies two or three decades ago, also contribute new technologies. Only a few of the largest mature companies conduct basic research, often under the pressure of product development needs. Their primary emphasis lies in anticipating and meeting the world's demand for large numbers of innovative, reliable, and affordable computer products of high quality.

The U.S. innovation engine works because its components complement one another. People and ideas flow among the three components of the engine, helping to integrate the intellectual curiosity of academia, the vigor and flexibility of the entrepreneur, and the resources and dependability of the giant corporations. This union is not perfect: companies and, in particular, computer science and technology graduate programs have complained of shortages in skilled computer science and technology personnel; college programs have suffered from obsolete equipment; university researchers are often slow to explore the real-life problems facing companies; the capital markets are too impatient, with a quick profit orientation that discourages risk taking and disparages long-term research; shakeouts among the venture companies often sweep away good ideas before they have any chance to pay off and too frequently reward the imitator rather than the innovator; large companies are often bureaucratic and reluctant to adopt new ideas. But, at least until now, the engine's strengths have clearly outweighed these weaknesses, and the ability of the United States to generate and bring to market a steady stream of innovations has been unparalleled.

THE RESEARCH INFRASTRUCTURE

An important element responsible for the successes of the innovation engine in both academia and industry has been the presence of an experimental infrastructure consisting of advanced research tools. Advanced research tools are as essential to the work of many computer scientists as particle accelerators have been to the work of high-energy physicists. Their importance is directly related to the largely experimental character of the discipline. Advanced technological systems will continue to be required by experimental computer scientists who are attempting to address research problems at the cutting edge of the field. Three important elements of this research infrastructure are discussed below.

Advanced Computer Resources for Research

Providing researchers with machines of the greatest possible speed and memory capacity and with the most advanced software systems has proven to be a sound investment in the future. In the past, dedicated large computers and forefront workstations enabled researchers to write larger programs, express them better, run them more rapidly, and advance the state of the art faster than they could have if they had had to rely on more limited personal computers or on the keyboards of dumb terminals attached to time-shared mainframe computers. Network connections have enabled researchers to share their results and gain access to important sources of information.

In the future, local computational resources for individual researchers will grow in number and improve in individual performance. We expect that massive multiprocessors and many of the other promising technological innovations already mentioned and further described in Part II will find their way into tomorrow's advanced workstations. The trend must continue if the U.S. arsenal of research tools is to be the best worldwide, enabling U.S. researchers to be among the first in making and consequently in exploiting new discoveries.

Prototyping Through Emulation and Simulation

A second important element of the research environment has consisted of powerful emulators and simulators—tools used to prototype ambitious software and hardware systems before they are built. Typically, these tools take a relatively long time (perhaps hours) to imitate how the system being analyzed would behave in a very short time interval (perhaps a fraction of a second). Nevertheless, they save time and money compared to the alternative of building and testing systems based on new and uncertain ideas. The increasingly complex architectures of contemplated systems, such as multiprocessors, speech and vision systems, and supercomputers, make this kind of pre-production modeling mandatory.

Improved VLSI Design and Fabrication

The third important component of the computer science research infrastructure has been the design and prototyping of new solid state circuits (VLSI chips) that are the building blocks of all computer systems. Improvements in VLSI architectures, i.e., in the design of these circuits in terms of more elementary components, and in the processes that translate these designs into silicon implementations are important because they lead to more powerful VLSI functions that, in turn, make ever more sophisticated computer applications possible. For example, a key innovation at the chip level giving rise to crucial and novel components of a special-purpose multiprocessor might, in turn, lead to higher-level systems capable of speech understanding, vision, and learning. VLSI improvements also have strong commercial implications. For example, the relatively low cost of the many home appliances that utilize microelectronics is a direct consequence of more powerful and less expensive VLSI components.

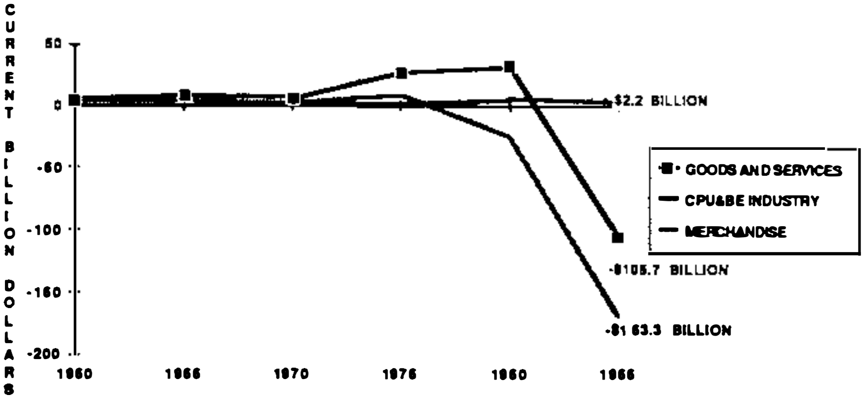
An important issue related to VLSI design has been the ability of the research community to convert its ideas into silicon circuitry as rapidly as possible. To date, researchers have used private companies along with the government-sponsored Metal Oxide Semiconductor Implementation System (MOSIS) foundry to that end with turnaround times from design to silicon prototype ranging from a few weeks to several months. Increasing dependence of computer systems on their components, the fast pace of change in computer technology, and growing foreign competition in semiconductor technology make it important to strengthen these foundry processes, speed up their turnaround times, make them more widely available, and maintain them at the cutting edge of research frontiers as a crucial component of the research infrastructure.

WHY THE ENGINE MAY NOT RUN SMOOTHLY IN THE FUTURE

The U.S. system has proved uniquely successful on a global scale. Other nations, including our strongest competitors, have different institutional structures, which, by and large, have not been as conducive to computer science innovation. That is, national differences have inhibited the duplication of our innovation engine. Japan, for example, has healthy mature companies that have generated impressive advances in the implementation of computer technologies, but its universities are weaker innovators and the country has virtually no venture sector. Western European companies, mature or venture, lack U.S. strength or consistency in bringing innovations to market, while the venerable Western European universities do not contribute as extensively to innovation in computer science as do their U.S. counterparts. This has not stopped the Europeans, however, from becoming very successful at innovating programming languages (such as Ada and Pascal).

In the absence of comparably effective institutions, other countries have tended to depend on innovations generated by U.S. research. This dependence, as well as widespread appreciation for the global scale of competition in computer-related markets, has led industrialized and newly industrializing nations to embark on programs involving government, industry, and/or academia to strengthen local computer science and technology capabilities. With consistent funding and commitment among the parties, these programs may eventually give rise to robust innovation engines tailored to different local societies. This development may take time, but many nations look more favorably than the United States does on investments with long payback periods. In the meantime, competitors from other nations who may follow U.S. firms in introducing new products benefit from lower expenditures on research, development, and market building. In this copy-cat environment, productivity and the many factors that give rise to competitiveness (such as product design, marketing, pricing, and quality) determine which competitors—and which countries—will be market leaders over the long run.

Foreign nations are becoming increasingly competent in computer science and technology, a development that may increase their productivity and competitiveness compared to those of the United States. In the aggregate, the U.S. trade surplus in computer and business equipment peaked in 1981 at just under $7 billion; by 1986, it had fallen to about $2.2 billion (CBEMA 1987; see Figure 2). If the pattern in computer science and technology were to follow that of textiles, steel, automobiles, and machine tools, we would lose a key source of competitive advantage in the world economy and a key source of national security. Already the Far East has achieved superior market share in commodity semiconductors (DRAMs) and personal computers (clones of U.S. machines) manufactured in the Pacific Basin (see Figure 3). While these products may not be state-of-the-art, Japan has moved aggressively into supercomputer production (it already produces some of the fastest machines available), and the West Europeans have made substantial advances in



SOURCE: U.S. Department of Commerce

FIGURE 2 U.S. trade balance versus computer and business equipment industry trade balance, 1960-1986. Reprinted, by permission, from CBEMA, 1987. Copyright © by the Computer and Business Equipment Manufacturers Association.

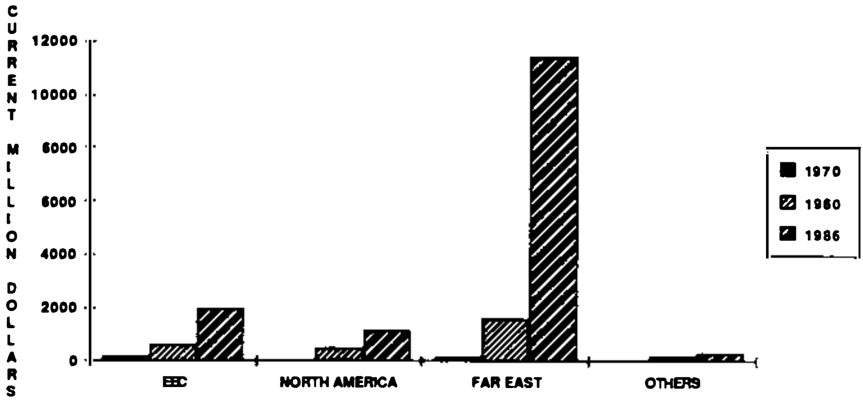


FIGURE 3 Computer and business equipment industry imports by geographic area, 1970-1986. Reprinted, by permission, from CBEMA, 1987.

software and systems integration. Further, Japanese strengths in consumer electronics are expected to facilitate computer technology development in such areas as optics, which is a focus of emerging storage technologies.*

The necessary development of the field and maintenance of the U.S. innovation engine are impeded by underrecognition of the need for basic research in computer science and technology. The situation is symptomatic of the more general problem in U.S. industry of not adopting a sufficiently long-term perspective, as has been revealed in the more established industrial sectors of chemicals and aircraft (positively) and steel, automobiles, and consumer electronics (negatively) (U.S. Congress, Office of Technology Assessment 1983; OECD 1985). While far-sighted companies may invest in the tools, facilities, and training required or their product strategies, U.S. industry generally does not tend to invest significantly in either basic research at universities or basic education. Such investments have long been the province of government, to which we direct our concerns.

Because basic computer science research depends so heavily on funding from two organizations—DARPA and NSF—it is particularly sensitive to changes in policy and funding behaviors at those organizations. For example, DARPA has recently begun to retreat from its traditional support of long-term basic research. Funding of basic computer science research has declined since 1983, while funding for applied research has more than doubled. DARPA has also begun to seek research results with more immediate military relevance and has instituted more bureaucratic procedures for funding and managing new projects. In part because the experimental nature of computer science research makes tight management difficult, this trend may undermine progress in the computer field.

Recently, NSF has attracted attention through its programs to launch supercomputer research centers and provide computer networking for researchers. These programs operate out of the same unit responsible for most of NSF's computer research funding; they use computer technology and employ computer scientists to assist researchers in the physical sciences, but they do not support basic research in computer science and technology. In contemplating the

---

*Competitiveness in computer science and technology depends on many factors and is sensitive to many influences. In recognition of the importance and complexity of the topic, the board is planning to focus projects on key issues pertaining to the competitiveness of the U.S. computer sector.

future of this area, it is important for policymakers to recognize that increasing the computer sophistication of the physical (and other) sciences, although extremely important, is not the same as conducting computer science research. Consequently, funding for scientific computing should be evaluated separately from funding for computer science research.

The NSF contribution to basic computer science research is less than half that of DARPA—approximately $60 million versus approximately $155 million. A major component of the NSF effort—the computer and computational research program—was funded at $19 million in 1987, grew only to $20 million in 1988, and is likely to grow by less than $2 million in 1989 in a highly constrained budget environment. Coming from the two principal federal supporters of this field, this order of magnitude for federal basic computer science research support, approximately $215 million, is alarmingly low. Total federal investment in basic computer science research, including high-performance computing research, is estimated by OSTP at only $300 million (OSTP 1987). Both figures may also be overestimates, since some applied research tends to be labeled basic research.

The board may study computer science and technology research patterns to derive more insight into current trends. But available evidence arouses its concern since, as noted below, pursuing even one of the grand challenges in the field may require more than $100 million dollars. In a budget environment that is driven by the immediate missions of government agencies and characterized by fragmentation of support among multiple agencies with multiple missions, computer science and technology appears to be at risk of losing support for basic research at a time when increased funding is needed more than ever before. As a relatively young field, with only the beginnings of the theoretical and empirical bases needed to achieve the substantial advances described elsewhere, computer science and technology is particularly vulnerable to the increasing politicization of federal research support.

5
Recommendations

The most important element of the challenge before us is the potential of computer science and technology to improve the nation's economic productivity and competitiveness. We recognize that improvements in these areas will not be achieved by technological progress alone; changes must also address a long list of economic, educational, managerial, labor, financial, trade, and policy issues that have been partially identified and are still the topic of current investigations and discussions. If progress in these areas is sufficient to maintain the United States at a level of rough parity with other nations, then national primacy in computer science and technology could supply a much-needed competitive edge.

Another significant element of the challenge of continued leadership in computer science and technology involves the national defense. The U.S. military depends heavily on computers in the operation of strategic, tactical, intelligence, logistic, and command and control systems and is supporting significant research on a number of new-generation computing technologies with potential military applications. Continued and enhanced leadership in computer science and technology ensures a healthier and stronger U.S. defense capability. Indeed, the vital dual role played by computers in defense and in the civilian economy plus their contribution to scientific research underscores that leadership in information technology is essential to U.S. national security.

The United States has no coordinating agency corresponding to Japan's Ministry of International Trade and Industry, and our nation has neither the appetite for nor the inclination toward centrally directed 5-year plans. If the U.S. government does not continue its established role in guiding and supporting basic research in computer science, no one else will. Government funding, particularly through DARPA, NSF, DOE, and NASA, is the de facto lever of U.S. national policy for setting the future direction of computer science and technology. Government also plays a role in shaping the application of computer science and technology as, for example, through tax incentives or military procurement.

We present below two broad recommendations that are aimed at helping the nation meet the national challenge in computer science and technology. The recommendations, which are mutually reinforcing, should be regarded as guides to strategic directions. In its ongoing and future work, the board will develop more specific recommendations. At this time, it is the board's purpose to underscore the importance of the issues and the need to assign high priority to addressing them.

IMPROVE AND EXPAND INFORMATION NETWORKING

Options should be evaluated for improving the computer communications infrastructure of the United States—providing for such features as fast, high-quality data transmission, support for a range of computer systems, ease of use, economical nationwide interconnectivity, and access to a range of information services and resources. At the same time potential costs and financing should be considered, along with risks, design alternatives and technology requirements, social and economic impacts, legal and regulatory aspects, and roles for industry and government. As the ongoing debate surrounding the more limited but component issue of improved computer networking for scientific researchers has shown, the issues associated with providing a nationwide service are numerous and complex (CSTB 1988). The board itself has begun and will continue to evaluate issues surrounding the information infrastructure. It recommends that the Congress and the Executive Branch also pursue such study as a first step toward developing appropriate policy.

As emphasized above, a substantially improved information infrastructure could help the United States to achieve many of the potential benefits promised but not yet necessarily achieved by computers, in particular those for improving productivity. The infrastructure could also accelerate implementation of many of the advances expected in computer science and technology, especially those conducive to network-based access or those that involve interconnection themselves. This promise, combined with the challenge to preserve U.S. leadership in the computer field, makes it imperative to explore information infrastructure options now.

SUPPORT FUNDAMENTAL ADVANCES IN COMPUTER SCIENCE AND TECHNOLOGY

Continued national strength in any industrial sector calls for sustained, long-term research and development. At a time when other nations are targeting and strengthening their own national research and development efforts in computer-related basic research—examples include Japan's SIGMA project and the European Economic Community's Esprit project (U.S. Department of Commerce 1988)—it is altogether appropriate, if not imperative, that the United States step up its own efforts.

We recommend that the strengthening of U.S. basic research in computer science and technology be achieved by identifying and funding grand challenges in the field, investing in human resources, strengthening the research environment, and increasing funding for basic research. Some of the relevant issues are discussed below.

Identify and Fund Grand Challenges

As in other fields of scientific endeavor, in computer science and technology there are a number of grand challenges worthy of long-term research support. Such challenges, if successfully met, would generate major advances in the field and create significant spin-offs in industry and government. We present here a partial list of such grand challenges, without suggesting that every item should be pursued. Instead, a few of these challenges combined with others of comparable magnitude from other lists should be selected and funded by government on a long-term basis.

1. Technology for large, correct software systems. The major component of the cost of computation is now the development and maintenance of software. Not only is the cost high, but so is the uncertainty: it is exceedingly difficult to know whether a piece of

software is correct, and development schedules and budgets often far underestimate actual costs. This challenge calls for tools, methods, and predefined components that will allow development at reasonable, predictable cost of large software systems (i.e., systems that require a million lines of code) that are knowably correct when they are released and that admit of modification at a cost proportional to the magnitude of change.

2. An ultra-reliable computer system. This challenge would entail development of hardware and software technologies for computer systems that could run 20 years or more on average without failing. Hardware-based fault tolerance requires research in redundant architectures at the component level. Software-based fault tolerance requires research in software-based dynamic reconfiguration of systems, involving concepts of virtual memories, virtual processes, and virtual data paths. Fault tolerance based on artificial intelligence requires research in intelligent monitoring, automated diagnosis, and repair. Ultra-reliable computers would benefit all applications, particularly in space and in hazardous, critical environments.

3. A trillion operations per second ultracomputer. Development of the so-called tera-ops computer system would require advances in a number of component technologies, such as a microprocessor with 1 to 10 nanoseconds cycle time, a billion bits of memory on a square inch chip, and a multiprocessor-memory-switch system with a trillion bits per second bandwidth. Achievement of such an ultracomputer would make possible novel scientific explorations and advances (see Chapter 6). To achieve this goal, research is needed in multiprocessing, parallel algorithms, switching, graphics, and memory management.

4. A translating telephone. This challenge calls for the development of a telephone by means of which people, speaking different languages, can converse directly. The Japanese have already undertaken this challenge by recently initiating a 7-year $120 million (16 billion yen) project toward developing a phone system in which a Japanese speaker can converse with an English speaker in real time. This challenge requires a speech system capable of recognizing large-vocabulary, spontaneous, unrehearsed, continuous speech; a natural-sounding speech synthesis approach that preserves speaker characteristics; and a natural language translation system capable of dealing with ambiguity, nongrammaticality, and incomplete phrases. Since achievement of a translating telephone would, in effect, be a simultaneous achievement of speech recognition, it would have a broad

and dramatic impact on human-machine communication throughout essentially all computer applications.

5. Specific systems that learn from practice. There has been a long and continuing interest in systems that learn and discover from examples, observations, and books. Currently, there is some research on systems that can learn from signals and symbols. Two longer-term grand challenges in this area are to develop computers that (a) can read a chapter in a college freshman text (say physics or accounting) and answer the questions at the end of the chapter, and (b) learn to assemble an appliance (like a food processor) from observing a person doing the same task. Both are extremely hard problems requiring advances in sensory computing, language, problem-solving techniques, and, most important, learning theory. Achievement of these challenges would open the door to systems that can learn by observation and practice and would therefore result in systems that improve continuously and adapt to change, thereby liberating us from the tedium and inflexibility of programming at the outset against all foreseeable eventualities.

6. Self-replicating systems. There have been a few theoretical studies in this area since the 1950s. The problem is of some practical interest in areas such as space manufacturing. Rather than launching a whole factory into space, it may be possible to send a small set of machine tools that can produce perhaps 95 percent of the parts needed for such a factory, using locally available raw materials and assembling the factory in situ. The solution to this problem involves many different disciplines. Research problems in this area include knowledge capture for reverse engineering and replication, design for manufacturability, and robotics.

Each of the above grand challenges would require and would generate significant breakthroughs and fundamental advances in computer science and technology. In each case, success or failure could be clearly established and appreciated by nonexperts. And each of these tasks would require long-term stable funding at significant levels. As we noted above, the Japanese are already budgeting some $120 million over the next 7 years on the translating telephone alone (and it is estimated that such other major Japanese computer research efforts as the Fifth Generation project and the Superspeed project will each cost more than $100 million annually (OSTP 1987)). Each of the other grand challenges on the list would probably call for funding at comparable or greater levels.

Success would by no means be guaranteed, but the payoffs from these efforts, even if they achieved far less than total success, would be substantial. Besides advancing the state of the art, the pursuit of such challenges would create a new generation of leading computer researchers, who in turn would contribute to the creative and effective use of computers throughout the nation. These challenges are grand, not only because of the immediate accomplishments sought, but because achieving them would give rise to immense technological spin-offs benefiting industry, defense, and society.

Invest in Human Resources

The single most important factor crucial to the success of U.S. computer science and technology is a reservoir of experienced, knowledgeable, and creative people. Developing and benefiting from this invaluable human resource requires action across several fronts to increase the number and enhance the quality of the people involved. For example, increasing the number of people well-educated in the productive use of computers depends on increasing the number of qualified teachers at all levels, beginning with elementary school. At the college level, computer science has the highest student-to-faculty ratio of any of the physical science and engineering disciplines (Gries et al. 1986). Because of the shortage of properly trained teachers, in many schools those teaching the subject are inadequately qualified.

It is especially important for graduate teaching and research that we increase the national supply of Ph.D. holders in this field, for which demand from educational institutions as well as industry has exceeded supply by an estimated 4 to 1 ratio (Hamlin, cited in Gries et al. 1986). In 1987, U.S. universities awarded only 466 computer science Ph.D.s, 2 percent of all Ph.D.s in science and engineering. Almost half of those degrees went to foreign students. By contrast, 845 Ph.D.s were awarded in mathematical sciences. The average computer science department has 18.5 faculty, while the average mathematics department has 30.2 (NSF 1987, Gries and Marsh 1988). Since the top-ranked universities are already producing Ph.D.s to the limits of their capacities, the pool of research talent can be expanded only by substantially increasing the funding for additional computer science departments and by funding efforts to improve the teaching of computer science and technology at all educational levels.

While the number of students entering graduate school in computer science is fortunately still increasing, there may soon be a downward trend. Undergraduate enrollments, after 5 years of dramatic annual increases and severe straining of human and physical resources, have declined over the past few years almost as precipitously as they had been increasing. Moreover, the quality of students pursuing college courses in computer science (and other sciences and engineering) appears to be declining. Students are not well-prepared and seem unwilling to undertake the hard work of mastering subjects with a mathematical basis. It is more important than ever that we catch the imagination of youngsters in school and create an awareness of the intellectual excitement and rewards of the field. We feel that this can best be done by teaching computation as a mode of thought as important as the disciplines of mathematics and the natural sciences. Student and teacher interest should also be stimulated early with hands-on experience in schools, business, and industry via summer jobs, internships, sabbaticals, and the like. To continue to cultivate a growing pool of researchers at the college level, significant increases in the number of graduate fellowships and other forms of financial support for both students and faculty are required.

Investing in people to become knowledgeable in computer science and technology goes beyond the satisfaction of basic research needs; it increases the talent available for more widespread use of computers throughout the economy. This is especially important in the areas of software development, with its severe mismatch between programmer demand and supply. The board will be examining human resource issues in computer science and technology in more detail to focus attention on problems and options for their resolution.

Strengthen the Research Environment

Computer science has a strong experimental component whose past progress has been intimately linked to the availability of advanced experimental resources. Accordingly, we recommend making possible the acquisition by research centers of advanced computer workstations, high-performance machines, multiprocessors, and other advanced computer technology research tools on an ongoing basis to guarantee that computer science researchers have at all times the best available tools to continue advancing the state of the art. We also recommend funding the use of advanced emulators and tools for prototyping to encourage and facilitate the design of innovative new architectures and software tools. Moreover, we recommend increased support for efforts in VLSI design and processes by maintaining and improving foundry services such as MOSIS. Finally, access to high-speed, high-functionality computer networks is critical for progress in this field, as in others.

Increase Funding for Basic Research

The board recommends that increased funding be directed to the promising technological areas outlined in Part I and discussed in more detail in Part II. Those areas fall within the categories of machines, systems, and software; artificial intelligence; and theoretical computer science. The board further recommends that special initiatives be undertaken to strengthen promising research that has been underfunded in the past, in particular in the areas of theory, software productivity, and commercial applications of computer technology and infrastructure. The latter two areas are directly relevant to U.S. productivity and competitiveness. In the course of its ongoing and future work, the board will address specific research needs in various segments of the computer science and technology field.

Part II
The Science, Engineering, and Technologies

In Part II we take up the scientific, engineering, and technological thrusts that were identified and summarized in Part I as most likely to influence the evolution of the field in the next decade and beyond. Here we provide more detail about their potential uses as well as some of the associated problems and prospects. Again, we remind the reader to keep in mind two important points: first, there are at least four technologies relevant to the link between computers and productivity—computer technology, communication technology, semiconductor technology, and packaging and manufacturing technology. We have focused on the first. Second, we are not presenting here an exhaustive taxonomy of all the computer subfields and their capabilities, but rather what we consider the most promising. The absence of discussion on areas such as databases does not mean that they are less important, but rather that they are likely to evolve further primarily through exploitation of the principal thrusts that we do discuss, such as those in multiprocessors and intelligent systems.

6

Machines, Systems, and Software

One of the biggest tasks facing computer designers is the development of systems that exploit the capabilities of many computers in the form of what are called multiprocessor and distributed systems. Related to current and future hardware systems is the task of developing the associated software. These three topics are discussed below.

MULTIPROCESSOR SYSTEMS

Multiprocessor systems strive to harness tens, hundreds, and even thousands of computers to work together on a single task, for example, to solve a large scientific problem or to understand human speech and transcribe it to text. These systems involve many processors located close to each other and with some means for communicating with one another at relatively high speeds.

The effort to build multiprocessor systems is the consequence of a powerful economic trend. Generational advances in VLSI design have made it possible to fabricate powerful microprocessors relatively cheaply; today, for example, a single silicon chip capable of executing 1 million instructions per second costs less than $100 to make. But the cost of manufacturing a very high-speed single processor, using a number of chips and other high-speed components, has not declined correspondingly. As a result, the world's fastest computers, which perform only hundreds of times faster than a single microprocessor, cost hundreds or thousands of times as much. Consequently, linking many inexpensive processors makes sound economic sense and raises the possibility of scalable computing, i.e., using only as many processors as are needed to perform a given task.

Another important incentive to build multiprocessor systems is related to physical limits. Advances in the speed of large, primarily single-processor machines have slowed down: after averaging a tenfold increase every 7 years for more than 30 years, progress is now at the rate of a threefold increase every 10 years (Kuck 1986). These superprocessors are approaching limits dictated by conflicts between the speed of light and thermal cooling: they must be small for information to move rapidly among different circuits of the machine, yet they must be large to permit dissipation of the heat generated by the fast circuits. Multiprocessors are expected to push these limits higher because they tackle problems through the concurrent operations of many processors. If the recent advances in superconductivity result in effectively raising these limits for superprocessors, they will also raise them for multiprocessors and thus the relative advantages of multiprocessors over single processors will remain the same.

The key technological problems related to the creation of useful mutliprocessor systems include: (1) the discovery and design of architectures, i.e., ways of interconnecting the processors so that the resulting aggregates compute desirable applications rapidly and efficiently; (2) finding ways to program these large systems to perform their complex tasks; and (3) solving the problem of reliability, i.e., minimizing failure of performance within a system in which the probability of individual component failures may be high. Current experimental work in the multiprocessor area includes exploration of different processor-communication architectures, design of new languages, and extension of popular older languages for multiprocessor programming. Theoretical work includes the exploration of the ultimate limitations of multiprocessors and the design of new algorithms suited to such systems.

The potential uses of multiprocessors are numerous and significant. The massive qualitative increase in computing power expected of multiprocessors promises to make these systems ideally suited to large problems of numerical and scientific computing that are characterized by inherent parallelism, e.g., weather forecasting, hydro- and aerodynamics, weapons research, and high-energy physics. Perhaps more surprisingly, conventional transaction-oriented computing tasks in banks, insurance companies, airlines, and other large organizations can also be broken down into independent subtasks—organized by account or flight number, for example—indicating that they can be managed by a multiprocessor system as well as, and potentially more cheaply than, by a conventional mainframe computer. In short, the fact that current programs are sequential is not a consequence of their natural structure in all cases, but rather of the fact that they had to be written sequentially to fit the sequential constraint of single-processor machines.

Most promising of all, multiprocessors are viewed by many computer scientists as a prerequisite to the achievement of artificial intelligence applications involving the use of machines for sensory functions, such as vision and speech understanding, and cognitive functions, such as learning, natural language understanding, and reasoning. This view is based on the large computational requirements of these problems and on the recognition that multiprocessor systems may imitate in some primitive way human neurological organization: human vision relies on the coordinated action of millions of retinal neurons, while higher-level human cognition makes use of more than a trillion cells in the cerebrum.

Traditional supercomputers, which rely on one or a handful of processors running at very high speeds, have already demonstrated their utility in several important applications. The proven capabilities of supercomputers will be greatly multiplied if the potential of multiprocessors is realized, leading to the tantalizing possibility of ultracomputers, which will harness together large numbers of superprocessors to yield mind-boggling computational power. Such power, in turn, could be used to expand scientific capabilities, for example, through computational observatories, computational microscopes, computational biochemical reactors, or computational wind tunnels. In these applications, massive-scale simulations would be performed to address previously unsolved scientific problems and to chart unexplored intellectual territory.

DISTRIBUTED SYSTEMS

Distributed systems are networks of geographically separate computers—collections of predominantly autonomous machines controlled by individual users for the performance of individual tasks, but also able to communicate the results of their computations with one another through some common convention. If a multiprocessor system can be likened to several horses pulling a cart with a single destination as their goal, then a distributed system can be likened to a properly functioning society of individuals and organizations, pursuing their own work under their own planning and decision schemes, yet also engaging in intercommunication toward achieving common organizational or individual goals.

Multiprocessor systems link many computers primarily for reasons of performance, whereas distributed systems are a consequence of the fact that computers and the people who use them are scattered geographically. Networking—making it possible for these scattered machines to communicate with one another—opens up the possibility of using resources more efficiently; more important, it connects the users into a community, making it possible to share knowledge, improve current business, and transact it in new ways, for example, by purchase and sale of information and informational labor. Networking also raises new concerns about job displacement; the potential for the invasion of privacy; the dissemination and uncritical acceptance of unreliable, undesired, and damaging information; and the prospect of theft on a truly grand scale. These are reminders that technology, like any innovation, carries with it risks as well as benefits and that safeguards must be provided to protect against such incursions. Devising appropriate safeguards is itself an urgent topic of theoretical systems research.

Distributed systems have emerged naturally in our decentralized industrial society. Their emergence reflects the proliferation of computers, especially the spread of personal desktop computers, and the appetite of users for more and more information. It also reflects the demands of the marketplace, in which users operate sometimes as individuals, at other times as members of an organization, and at still other times for interorganizational purposes. Distributed systems rely on a range of communication technologies and approaches—including telephone and local area networks, long-haul networks, satellite networks, cellular, packet radio and optical fiber networks—to connect computers and move information as necessary.

Distributed systems are at the basis of modern office automation. They are evident in computer networks such as the ARPANET, which has facilitated the exchange of ideas and information within the nation's scientific community. Perhaps most significant, distributed systems have begun to transform national and international economic life, creating the beginnings of an information marketplace geared to the exchange of information. Electronic mail enables communities of users to annotate, encapsulate, and broadcast messages with little or no handling of paper and makes it possible to send people-to-people or program-to-program messages. Customers from every part of the country can tap into centralized resources, including bibliographic databases, electronic encyclopedias, or any of a growing number of specialized financial, legal, news gathering, and other information services. Geographically separated individuals can pool resources for joint work: a manual for a new product can be assembled with input from the technical people on one coast and from the marketing staff on the other; a proposal can be circulated widely for comments and rebuttals from many contributors electronically. Homebound and disabled individuals can participate more actively in the economy, liberated by networking from the constraints of geographical isolation or physical handicap. The technology of distributed systems, through enhanced and nationwide network access, could have a major and unique impact on the future economy.

The limitations of today's distributed systems, whether they form a small local system in a building or a much larger corporate communication system, inhibit the growth of an information marketplace. They do so because the systems communicate at a rather low level, with the only commonly understood concepts being typed characters and symbols. They also are often heterogeneous, made up of machines from a variety of manufacturers, which employ a number of different hardware and software conventions. Except at the lowest level of communicating characters, there are no universally accepted standard conventions (protocols), although such shared communication regimes represent the first level of software needed for providing a higher level of commonality of concepts among such disparate machines. As a result, if two computer programs are currently to understand each other in order, for example, to process an invoice, they must be specifically programmed to do so. Such understanding is not shared by other number of machines unless they are similarly programmed. The greater the number of machines participating in a distributed system, the greater the agreement needed on common programming. Such agreement is not easy to arrive at, in part because system heterogeneity makes it difficult to implement. Consequently, one of the major problems ahead is the development of common and effective distributed system semantics—that is, languages and intelligent software systems that will help machines communicate with and understand one another at levels higher than the communication of characters, and whose use is simple and fast enough to win acceptance among a large number of users.

Despite these obstacles, the number of interconnected systems is increasing because of their great utility. The ongoing proliferation of these computer networks provides a test bed for research and a powerful incentive for developing an understanding of their underlying principles. Beyond the need for distributed system semantics, other important aspects include the development of innovative and robust system architectures that can survive computer and communication failures without unacceptable losses of information, the development of network management systems to support reliable network service on an ever growing scale, and the creation and evaluation of algorithms specifically tailored to distributed systems. Finally, on the software side of these systems, the problems associated with programming large and complex computer systems must be better understood.

SOFTWARE AND PROGRAMMING

Computers are general-purpose tools that can be specialized to many different tasks. The collections of instructions that achieve this specialization are called programs or, collectively, software. It is software that allows a single computer to be used at various times (or even simultaneously by several users) for such diverse activities as inventory and payroll computations, word processing, solving differential equations, and computer-assisted instruction. Programs are in many ways similar to recipes, game rules, or mechanical assembly instructions. They must express rules of procedure that are sufficiently precise and unambiguous to be carried out exactly by the machine that is executing them, and they must allow for error conditions resulting from bad or unusual combinations of data.

Unlike other products, the essence of software is in its design, which is inherently an intellectual activity. This is so because producing many instances of a program involves straightforward duplication rather than extensive fabrication and assembly. Accordingly, the cost of producing software is dominated by the costs of designing it and making certain that it defines a correct procedure for performing all of its desired tasks—costs that are high because of the difficulties inherent in software design.

Creating software involves devising representations for information called data structures and procedures called algorithms to carry out the desired information processing. One of the major difficulties associated with this task is that there are generally many different combinations of data structures and algorithms that can perform a desired task effectively. For example, in a machine vision system that inspects circular parts, a circle could be represented by its center and radius (a data structure of two numbers for every part) or by a few thousand points that approximate the circumference; the first data structure is more economical but does not by itself permit deviations from circularity to be represented, as does the second. In carrying out the artful process of data structure and procedure selection, the programmer must often pay equal attention to both large and small software parts, like an architect who must design a house, down to all its windows, doors, doorknobs, and even bricks.

Furthermore, it is difficult for programmers to anticipate during design all the circumstances that might conceivably arise while a program is being executed. Indeed, another difficulty involves the illusion that software, because it is the stuff of design, is infinitely malleable and can therefore be easily changed for improvement or to meet new demands. Unfortunately, software designs are often so complex and variations among different parts so subtle that the implications of even a small change are hard to anticipate and control. These factors make it fundamentally hard to specify and design software. They also make software difficult to test by anticipation of all the failure circumstances that may accidentally arise during actual operation. For these reasons, software development is often quite costly and time consuming.

Beyond design, and after a program is put to use, modifications are often required to repair errors, add new capabilities, or adapt it to changes in other programs with which it interacts. This activity is called software maintenance—a misleading term since it involves continued system design and development rather than the traditional notion of fending off the ravages of wear and age. Such maintenance can amount to as much as 75 percent of life-cycle cost (Boehm 1981).

In the 1960s the cost of computing was dominated by the cost of hardware. As the use of computers became more sophisticated, the cost of hardware dropped, and the salaries of programmers increased, software costs came to dominate the cost of computing (OECD 1985). The problem has been aggravated by an apparent shortage of good software professionals and limited productivity growth. The overall annual growth of programming productivity is at best 5 percent (OECD 1985). The increase in software costs is taking place throughout the field, from small programs on personal computers to life-critical applications and supercomputing. The cost increase is fueling efforts to improve software engineering through development of tools to partially automate software development and techniques for reusing software modules as parts of larger pieces of software. A lot of work is needed to increase productivity in software development by a factor of ten, considered a critical milestone in industry.

Another serious software problem has to do with people's perceptions and expectations. Hardware and software sound similar, and people are frequently appalled that as the former gets cheaper by some 30 percent per year, the latter stubbornly resists productivity improvements. Such a comparison, however, reflects a misunderstanding of the nature of the software development process; a brief recounting of the development of MACSYMA, one of the earliest knowledge-based programs, suggests why. To develop a research prototype of that program—a mathematical assistant capable of symbolic integration, differentiation and solution of equations—took 17 calendar years and some 100 person-years. In terms of the number of moving parts and their relationship to one another, the program's complexity was comparable to that of a jumbo jet, whose design and development cost more than 100 times as much. Most people can intuitively grasp the difficulties of constructing complex physical systems such as jumbo jets. But the complexities and design difficulties in the more abstract world of software are less obvious and less appreciated.

Despite all these difficulties, software development has seen significant progress. In the early days of programming, it was often a triumph to write a program that successfully computed the desired result. There was little widespread systematic understanding of program organization or of ways to reason about programs. Algorithms and data structures were originally created in an ad hoc fashion, but regular use and research led to an increased fundamental understanding of these entities for certain problem domains: we can now analyze and compare the performance of several proposed algorithms and data structures and, for several kinds of problems, we often know in advance theoretical limits on performance (see Chapter 8). Sound theories have also contributed to the construction of certain classes of software systems: for example, in the early 1960s the construction of a compiler (a program that translates programs written at a higher-level language to machine-level programs) was a significant achievement for a team of programmers; such systems are now constructed routinely (with largely automated means by a much smaller group) for traditional single-processor machines.

In today's computing environment, escalating demands for overall computer performance become escalating demands for software capability and software size. Development of large systems requires the coordination of many people, the maintenance and control of many versions of the software, and the testing and remanufacture of new versions after the system has been changed. The problems associated with these activities became a focus of computer science research in the mid-1970s, through techniques of modular decomposition (Parnas 1972) and organization of large teams of programmers (Baker 1972). At that time, a distinction was made between programming-in-the-small and programming-in-the-large to call attention to the difference between the problems encountered by a few people writing simple programs and the problems encountered by large groups of people constructing and managing sizable assemblies of modules.

Beyond these more or less pure software systems that deal only with information, there are other, even more complex, highly distributed systems that often interact with physical processes, such as the U.S. telecommunications, air traffic control, transportation, process control, energy, air defense, strategic offense, and command-control-communication and intelligence systems. These supersystems, as they have been called (Zracket 1981), grow over a period of decades from initially limited objectives to evolutionarily mature end states that are generally unpredictable at the start. The need to create such supersystems and other software with sufficient reliability for effective use presents a set of software design and development problems that are not addressed by the techniques of either programming-in-the-small or programming-in-the-large.

Accordingly, two new tasks for software research are: (1) to develop better techniques for designing software, especially software to be embedded in very complex, real-time application systems, and (2) to use emerging artificial intelligence techniques for the development of tools that will help software developers manage the complexity of such software. Other directions and opportunities for future software progress include: (3) effective ways of improving the productivity of the software development process, e.g., through automation of software design, reuse of existing software components, or new types of software architectures; (4) ways to reason about the correctness of software, including the task specification process; (5) infrastructural tools and resources, such as electronic software distribution systems; and (6) addressing the new multiprocessor software problems that will inevitably arise from the new multiprocessor architectures discussed above.

7

Artificial Intelligence

Artificial intelligence (AI) looms large in the public's perception of the future of computer science and technology and has contributed much to the emergence of this field. In this chapter, we focus on what we consider to be particularly promising aspects of AI: sensory computing, expert systems, deeper cognitive systems, and robotics.

SENSORY COMPUTING

Understanding the workings of the human sensory apparatus and implementing comparable capabilities on machines, particularly speech and vision, is an important scientific challenge and a technological imperative. It is vital to the development of autonomous devices such as robots and for improved communication between machines and their human users.

In the area of speech understanding, it has proved to be more difficult than expected to get computers to recognize untrained human speech. At present, systems can recognize a limited number of words, they take a relatively long time to do it, and speakers must usually pause between words. Even this modest success requires the speaker to familiarize the computer with the unique qualities of his or her voice by reading aloud lists of all the words to be used. For such speaker-dependent systems, the machine, after training, can achieve word recognition of several thousand words with a success rate in

the upper 90 percent range. Speaker-independent systems that can understand continuous speech appear to be feasible but are at least 3 to 5 years away. Advances in natural language understanding and cognitive science, combined with the potential of multiprocessor systems to provide the huge processing power required, hold out a big promise but not a guarantee of expanded capabilities for speech comprehension via computer.

Machine vision represents another critical area in which, as in speech, significant progress is likely to depend on the combination of cognitive research with the evolution of massively parallel and most probably special-purpose multiprocessor systems. Machine vision is the process of deriving useful information about a scene from images, for example, the conversion of a huge list of numbers representing the light intensities of millions of minute dots, which make up an overall picture as perceived by a video camera, into a description of the pictured objects, their location, and spatial relationships. This description may be used, in turn, to control a manipulator that picks up an object or to guide a vehicle on a road.

Since as long ago as 1950, demonstrations of machine vision have included recognition of printed characters, medical image analysis (e.g., counting blood cells), some industrial vision (e.g., printed circuit board inspection), flexible assembly, and military target detection. Despite these successes, however, the capabilities of machine vision today are still largely limited to printed character recognition, medical image analysis, and some industrial inspection. This is due in part to the low computational power available and in part to the youth of current theoretical foundations and algorithms that address visual perception.

Advances in machine vision are expected to have a significant impact over a large number of uses for the same reason that our own eyes are so important in everything that we do. Autonomous systems, be they military vehicles or robots on the factory floor, will have vision with far greater capability and flexibility than today's repetitive-motion robots. Another important consequence of improved machine vision and better speech comprehension will be the evolution of more natural interfaces that span nearly all applications of computers and permit users to speak or show things to their machines as naturally as they do in interacting with people.

EXPERT SYSTEMS

Expert systems involve techniques for representing knowledge and methods by which that knowledge can be used by a machine to reason toward the solution of problems that are difficult enough to require significant human expertise for their solution.

Every expert system consists of three principal parts: the knowledge base, the reasoning or inference methods, and their interface with the user. Knowledge bases contain factual knowledge and heuristic knowledge. The factual knowledge, like the knowledge in textbooks or journals, is widely shared and easily obtained. In contrast, the heuristic knowledge is rarely discussed and is largely in the private domain of experts. It is the knowledge of good practice, good judgment, and plausible reasoning in the field. It is the knowledge that underlies "the art of good guessing."

The inference methods used by expert systems are often based on propositional calculus or predicate logic. Most commonly used are "forward chaining" methods, which follow causal paths from conditions presented to the program to conclusions reached by the program (modus ponens applied repeatedly), or "backward chaining" methods, which proceed from goal statements to conditions (same logic backward). Probabilistic frameworks and some ad hoc frameworks are also used for inference.

As one would expect from a technology so broadly conceived, the span of applications is as wide as the world of professional and semiprofessional work. The earliest applications of expert systems were in such esoteric areas as the analysis of chemical data, medical diagnosis and therapy planning, the interpretation of data from oil well logging, and the defense-related interpretation of deep-ocean sound. As the applications of expert systems began to grow in the mid-1980s, other mainline commercial and industrial applications began to emerge. Finally, in government, expert systems are used to assist government officials in interpreting health care management data and complex pension laws. In mid-1987, there were approximately 1,500 applications in use and several thousand under development (Feigenbaum et al. 1988).

As expert systems continue to evolve, it is becoming apparent that two applications areas—manufacturing (in particular, the white-collar aspects) and financial services—are beginning to dominate. In each of these, the sheer economic volume of goods and services means that even small enhancements to the average human professional skill in decision making is leverage for great economic gain. Examples of expert systems in manufacturing include: the design of a manufacturable configuration of subsystems, given a customer order for a minicomputer, and the design of an associated floor layout; real-time scheduling and rescheduling (due to a machine failure) of the progress of wafers-in-process in a large microchip manufacturing facility; and planning the manufacturing process for jet fighter parts. In finance, expert systems are used to assist bank officers in deciding the credit worthiness of a loan applicant and to assist insurance underwriters in deciding price and terms for insurance contracts.

Probably more than half of today's expert systems are used for diagnostic purposes, such as assisting auto mechanics in diagnosing and repairing subsystems of automobiles and carrying out real-time remote diagnostic tests of massive steam turbine generators. Applications to diagnosis will continue to be widespread. Motivating this is the increasing complexity of devices and systems used throughout industry. Unassisted human abilities in problem solving, training, and retraining cannot keep pace with current and expected developments.

There are a number of key research issues in expert systems. (1) Knowledge representation: How shall the knowledge of a domain of human endeavor and the world in which it is situated be represented as data structures in the memory of a computer? (2) Knowledge utilization: How can this knowledge be used for problem solving? Essentially, this is the question of the design of inference (reasoning) procedures and frameworks. (3) Knowledge acquisition: How will it be possible to acquire the knowledge automatically (machine learning) or at least semiautomatically (transfer of expertise from humans, their texts, or their data)? (4) Large knowledge bases: The power of expert systems resides in the specific knowledge of the problem domain, and for systems to be powerful they must contain a large amount of high-quality knowledge. Accordingly, an enormous knowledge infrastructure needs to be codified and represented for machine use and, as one would expect, this is and will continue to be a huge endeavor in which machines may participate, as in (3) above.

Applications of precise knowledge delivery are also of increasing importance. A knowledge delivery application is one in which the right knowledge, in the context of a problem or a service, is delivered at the right moment for a human professional to consider. For example, one commercially available knowledge delivery system advises clinical pathologists about tissue diseases and associated features. Such applications are motivated by the great complexity of human systems and procedures that are now in place—a complexity that even the mind of a specialist cannot encompass. A knowledge delivery application is, in essence, a "living rulebook or textbook" that delivers knowledge in context.

As we look toward the future, the volume of expert systems is expected to grow and blend with the great stream of more conventional data processing and numeric applications. There is a certain inevitability at work here. As the cost of computers continues to fall during the coming two decades, many more of the practitioners of the world's professions will be persuaded to turn to information processing for assistance in managing the increasing complexity of their daily knowledge-related tasks. The computers that will act as intelligent assistants for these professionals will have to have reasoning capabilities and knowledge.

In time, we will undoubtedly achieve a broad reconceptualization of what is meant by an expert system. In the broader concept, the system will be conceived as a collegial relationship between an intelligent computer agent and an intelligent person (or persons). Each will perform tasks that it/he does best, and the intelligence of the system will be a result of the collaboration. Issues and problems expected to dominate the agenda of future expert system researchers include: (1) the creation of more powerful, general, and easy-to-use programming systems that will liberate the user from knowledge engineering intermediaries; (2) new knowledge representation formalisms and techniques, adequate and effective for representing a broad body of general knowledge about the everyday world, the worlds of science and engineering knowledge, biological and medical knowledge, and so on; (3) new reasoning methods that escape the elegant but rigid bounds of propositional and predicate logic and reuse old knowledge for solving new problems—forerunners of such methods are now called reasoning-by-analogy, case-based reasoning, script-based reasoning, and chunking; and (4) new machine learning methods for acquiring knowledge based on analogies, on abstractions from internal problem-solving processes, on watching human expert problem solving, and on the automated reading of textual material from journals and textbooks.

We can envision that as society changes from industrial to post-industrial and as work becomes increasingly the work of professionals and knowledge workers, the power tools will be expert systems. The economic and social well-being of advanced societies increasingly will be the result of "working smarter" rather than "working harder," and expert systems will be agents of that change. Knowledge is power in human affairs, and expert systems are amplifiers of human thought and action.

DEEPER COGNITIVE SYSTEMS

Another important focus in AI research involves the attempt to understand and model the deeper cognitive activities fundamental to intelligence, including learning, explaining, planning, and hypothesizing. Research in this area is an interdisciplinary enterprise, involving a synthesis of concepts from experimental psychology, linguistics, neuroscience, and computer science; advances hold the dual promise of increasing understanding of human cognitive processes and introducing more and more intelligence into the computer. Listed below are some of the promising current thrusts of this research.

1. The organization of memory. Work in the cognitive AI field has overturned the once-prevalent view that human memory could be viewed as a largely unorganized mental filing cabinet. AI researchers have developed several sophisticated and influential models of how humans organize their knowledge. Although these theories, which include semantic networks, frames, and scripts, involve different methods of representing the memory's organization, they share a common assertion that memory structure consists of a network of stored associations, with various types of information stored at each node of the network. Collectively these memory models have helped in the construction of knowledge-based systems that use contextual information to tackle specific problems.

2. Learning from practice. After a long lull caused by disappointments with early experiments on learning machines some 20 years ago, recent advances in the development of computer systems have given rise to programs that exhibit modest yet continuous learning from practice on the tasks that they perform, much as humans do. These recent innovations have important implications for computer science in that they represent key steps toward the goal of making more intelligent machines. Multiprocessors add fuel to this promise with the substantially greater power they possess. Machines that could learn from practice, even at a modest scale, could relieve much of the burden of programming all the necessary intelligence at the outset and could help tailor generic programs to specific applications.

3. Connectionism. The last decade has witnessed progress in the development of systems and theories involving the connectionist

paradigm, which is often likened to the human nervous system. The result of an interdisciplinary effort by neuroscientists, psychologists, and computer scientists, connectionist work grows from the shared conviction that the computational architecture of human cognition is fashioned within the highly parallel dynamic architecture of the human brain. Connectionist systems involve interconnected networks of large numbers of elemental computing nodes that often simply add up the values of their inputs and check if the sum is above a preset threshold. These massively parallel systems, sometimes referred to as neural networks, operate by learning strategies that involve the modification of the elemental nodes (e.g., the thresholds) in response to what they experience.

Recent advances in this area are related to new knowledge about what can be learned by such networks and improvements in VLSI circuits that make possible new complex architectures of many such interconnected cells. Progress has been constrained by the learning limitations of small experimental systems and by the absence of a theory sufficiently developed to address how large neural networks can become capable of substantial, predictable, and scalable learning.

Making systems more intelligent is a primary goal of AI research, and advances toward this objective will make computers both more useful and easier to use. The power and flexibility of today's machines are greatly inhibited by the amount of detailed knowledge that must be memorized by those who wish to use them effectively. Given advances in equipment and a deeper theoretical understanding of human cognitive processes, tomorrow's computers should have a greatly enhanced capacity to understand what unsophisticated users want them to do. More successful cognitive computer systems will enhance the usefulness and ease of use of computer systems in all areas of application.

ROBOTICS

Robotics researchers strive to understand and build machines that are sufficiently intelligent to interact effectively with the physical world in the performance of designated tasks. Progress in robotics will continue to be a key to enhanced productivity in factories, with particular utility in the performance of repetitive and dangerous jobs or jobs that require sustained quality control. Moreover, extensive use of robotics and other computer technologies in design and manufacturing is expected to make possible the rapid prototyping of products (or even factories), permitting the cost-effective manufacture of customized products at mass production costs. Autonomous systems with mobile and perceptual capabilities will also make possible the performance of otherwise-impossible tasks, such as planetary exploration over a long period of time. Listed below are some of the major promising research thrusts in robotics.

1. Sensors and perception. Sensors are the mechanisms that provide information about the robot's relation to the environment. Perception enables a machine to comprehend and adjust to its physical surroundings. Modes of sensing and perception include the visual (see Chapter 7), tactile, force, torque, speed, and even olfactory modes. Improvements in sensing and perception have the potential to augment the usefulness of virtually every type of machine by increasing its ability to adapt to the complexity and variability that characterize the physical world. An important research activity in this area is combining sensory transducers with computation for making smarter sensors. Machine vision represents the most important perceptual capability of future robotic systems.

2. Mechanisms. Progress in robotics depends on the design and manufacture of mechanisms capable of the subtle, strong, and precise motions required for useful activity. The need for precision, speed, light weight, and strength poses serious problems that cannot easily be met with conventional approaches. To date, most of the best work in this area has come from the intensive efforts of design teams relying on traditional engineering methods. We expect that extensive computer-assisted design will play an expanded role in this area through modeling and simulation of complete robotic mechanisms before construction.

3. Sensorimotor integration. To achieve smooth, flexible, efficient motions in robots, the sensor and motor controls must be integrated and coordinated. Advances here call for research on visual, force, tactile, and torque feedback. A deeper understanding of this integrative process among robot sensors and actuators will broaden understanding of neuroscience and biomechanics as well.

4. Planning. The effectiveness of robotics depends heavily on a machine's ability to define necessary actions and specify their sequence in order to achieve a desired goal. Planning ranges from high-level task planning (e.g., to assemble a product) to low-level path planning (e.g., for obstacle avoidance). A difficult and important problem in planning involves the conversion of semantic or mission descriptions of a robot's goals to physical or machine-executable

functions. Planning must also account for the inherent uncertainty and partial knowledge that robots have of their physical environment. Some researchers believe that the best hope for progress in planning rests with the creation of more intelligent programs with a deeper knowledge of the physical world.

8

Theoretical Computer Science

Computer science is a young discipline, and its theoretical base is immature. Potential applications of computers even to known problems will be limited until that base of theory is more fully developed. Theoretical computer scientists are concerned with the intrinsic limits to solving problems.

Since computer science is an artificial science (Simon 1981), theoretical computer science plays a very different role within computer science than, say, theoretical physics plays within physics. Theoretical physicists seek to understand the physical universe, which exists independently. Theoretical computer scientists seek to understand all possible architectures or algorithms, which computer scientists create themselves.

The concepts and results of theoretical computer science have influenced many disciplines. In particular, scientific computing, which has been referred to as the third leg of science (together with experimental and theoretical science), uses algorithms developed by computer scientists. New algorithms have been as important as advances in technology in our progress in solving many important problems during the last four decades.

COMPUTATIONAL COMPLEXITY

Some of the most influential theoretical work in computer science has involved the notion of computational complexity, i.e., the intrinsic difficulty of solving a given problem, independent of the particular algorithm employed to solve it. For example, some problems are known to require computational resources that grow linearly or quadratically with input size, while the resources required by other problems grow exponentially or in unknown ways. The theory developed to date enables computer scientists to demonstrate that various problems are easily solvable or provably intractable and further enables them to determine that numerous problems have essentially the same complexity. Computational complexity thus serves much the same function in computer science that the laws of thermodynamics play in the physical sciences (Packel and Traub 1987); it strives to determine the limits of the possible. An important open problem in theoretical computer science is to decide whether an important class of computational problems (the so-called NP-complete problems) really are intractable or simply resistant to the approaches used to date.

An example of an NP-complete problem is the traveling sales representative problem, which is formulated as follows: one is given a set of *n* cities together with the distances between all pairs of cities. A trip must be made that visits each city exactly once and returns to the starting point. The problem is to find a tour, that is, an ordering of the *n* cities, that minimizes the total distance traveled. This problem turns out to be an abstraction of many scheduling and layout problems in the real world.

A characteristic of the traveling sales representative problem and of many other well-known problems, such as linear programming, is that the available information is complete and exact. In principle, these problems can be solved exactly. Information-based complexity studies the intrinsic difficulty of problems for which the information is partial, contaminated by error or approximation, and provided at some sort of cost. These are characteristics of many problems in the physical sciences, the social sciences, and engineering. Such problems cannot be solved exactly, and one must live with uncertainty. Questions considered by information-based complexity include a problem's intrinsic uncertainty, the minimal computational resources required to achieve any chosen level of uncertainty, and whether the information can be decomposed for parallel or distributed computation.

ALGORITHMS AND THEIR ANALYSIS

An area in which theory has had significant practical impact in the past and is likely to have more in the future involves the search for new procedures to solve difficult problems.

While some computer scientists point to the dramatic increase in computational power as the development that has contributed the most to extending the computer's problem-solving utility, others argue that theoretical advances in the design and analysis of algorithms have had and will continue to have an even greater impact. Better understanding of how best to design and measure the efficiency of computational procedures, in terms of both speed and the amount of memory required, has resulted in numerous algorithms of practical utility.

The development of the fast Fourier transform, for example, produced an algorithm that is ubiquitous in the areas of signal and image processing and is used extensively in speech research, radar, sonar, oil exploration, and seismography. Indeed, the algorithm might be said to have made such processing possible; without it, the problems would be too computationally intensive to solve, hardware advances notwithstanding.

The development of a highly efficient sorting algorithm has also led to a broad range of applications, and a more recently developed linear programming algorithm has offered a solution to many common scheduling, distribution, and resource allocation problems. Current theoretical work in algorithms addresses a large set of similar practical problems and has moved increasingly into the realm of parallel and distributed systems.

SEMANTICS AND LANGUAGES

Theoretical work on computer languages has focused on syntax—how to generate and describe a language's grammar—and on semantics—how to determine the meaning of a program through the meaning of the operations it causes to be executed. Fundamental advances in these areas have drawn on mathematical logic to produce a conceptual framework that facilitates computer language translation on conventional architectures. Twenty-five years ago, for example, the generation of a Fortran compiler was considered a legitimate research topic, involving 20 to 50 person-years of work; today, compilers for advanced languages on conventional uniprocessors are routinely constructed by students over the course of a single academic semester.

Formal language theory developed in the 1960s and 1970s enabled the development of utility programs that can now automatically generate lexical analyzers and parsers from the grammar of a programming language. Theoretical developments on formal semantics in the 1970s and early 1980s are now enabling the automation of other aspects of compiler construction. Results from graph theory have improved the ability of compilers to optimize object code.

This dramatic improvement is attributable to a better understanding of computer languages and other computer science issues and suggests the potential practical results that can flow from future theoretical advances. At the same time, new and evolving architectures, especially multiprocessors, are presenting designers of compilers with new challenges and long development times, thereby attesting to the field's continued growth.

CRYPTOLOGY

Although code-making is an ancient craft, dating back to the Roman Empire, computer science theory has played an essential role in adapting encryption to meet the special demands of complex modern society. It is one thing to maintain privacy and security against a small number of potential eavesdroppers when one is communicating a limited number of messages to a limited number of recipients; it is quite another to preserve secure communications in an environment of vast, interconnected communications networks with thousands or even millions of users. Computer scientists have developed a technique by which the potential recipient of messages can broadcast a (disguised) password publicly (while retaining it privately) and be assured of being the only one able to decipher the encrypted messages received. The same approach can be used to distinguish the signatory of an electronic memo from an impostor. Such advanced encryption techniques have a narrow but important range of applications in both the commercial and national security areas.

References

Baker, F. T. "Chief Programmer Team Management of Production Programming." *IBM Systems Journal*, 11(1), 56-73, 1972.

Boehm, B. W. *Software Engineering Economics.* Prentice-Hall, Englewood Cliffs, N.J., 1981.

CBEMA. *The Computer, Business Equipment, Software & Services, and Telecommunications Industry 1960-1996.* The Computer and Business Equipment Manufacturers Association. Washington, D.C., 1987.

Computer Science and Technology Board, National Research Council. *Toward a National Research Network.* National Academy Press, Washington, D.C., 1988.

Cusumano, Michael A. "The 'Factory' Approach to Large-Scale Software Development: Implications for Strategy, Technology, and Structure." Working Paper #1885-87 (revised), MIT Alfred P. Sloan School of Management, Cambridge, Mass., September 25, 1987.

Dertouzos, M. L., R. K. Lester, R. M. Solow, and L. C. Thurow. "Interim Results of the MIT Commission On Industrial Productivity." American Association for the Advancement of Science Annual Meeting, February 15, 1988.

Feigenbaum, E. A., P. McCorduck, and H. P. Nii. *The Rise of the Expert Company.* Times Books, 1988 (forthcoming).

Gries, D., and D. Marsh. "The 1986-1987 Taulbee Survey Report: The Computing Research Board's Survey on the Production and Employment of Ph.D.'s and Faculty in Computer Science and Engineering." No. 88-897. Department of Computer Science, Cornell University, February 1988.

Gries, D., et al. "Imbalance Between Growth and Funding in Academic Computer Science: Two Trends Colliding." *Communications of the Association for Computing Machinery*, 29(9), September 1986.

Harmon, P. "Digital Equipment Corporation Saves Millions With Expert Systems." *Expert Systems Strategies*, 3(8), August 1987.

International Data Corporation. *Software and Services Industry Review and Forecast - 1986*. Association of Data Processing Service Organisations, Washington, D.C., May 1986.

Kuck, D., E. Davidson, D. Lawrie, and A. Sameh. "Parallel Supercomputing Today and the Cedar Approach." *Science*, 231, 967-974, February 28, 1986.

McCormick, B. H., et al., eds. *Visualization in Scientific Computing*, Report to the National Science Foundation by the Panel on Graphics, Image Processing and Workstations. Association for Computing Machinery Special Interest Group on Graphics, New York, July 1987.

National Research Council. *Renewing U.S. Mathematics: Critical Resource for the Future*. Report of the Ad Hoc Committee on Resources for the Mathematical Sciences. National Academy Press, Washington, D.C., 1984.

National Science Foundation. *Science and Engineering Indicators—1987*. Washington, D.C., 1988.

Office of Science and Technology Policy. *A Research and Development Strategy for High Performance Computing*. Executive Office of the President, November 20, 1987.

Organisation for Economic Cooperation and Development. *Software: An Emerging Industry*. OECD, Paris, 1985.

Packel, E. W., and J. F. Traub. "Information-based Complexity." *Nature*, 328(2), July 1987.

Parnas, D. I. "On the Criteria for Decomposing Systems Into Models." *Communications of the ACM*, 15(12), 1053-1058, December 1972.

Personick, V. A. "A Second Look at Industry Output and Employment Trends Through 1995." *Monthly Labor Review*, 26-41, November 1985.

Quinn, J. B. "The Impacts of Technology in the Services Sector." In B. Guile and H. Brooks, eds., *Technology and Global Industry: Companies and Nations in the World Economy*. National Academy Press, Washington, D.C., 1987.

Simon, H. *The Sciences of the Artificial*. The MIT Press, Cambridge, Mass., 1981.

U.S. Congress, Office of Technology Assessment. *Computerized Manufacturing Automation*. GPO, Washington, D.C., 1984.

U.S. Congress, Office of Technology Assessment. *International Competitiveness in Electronics*. GPO, Washington, D.C., November 1983.

U.S. Department of Commerce. *1988 U.S. Industrial Outlook*. GPO, Washington, D.C., January 1988.

U.S. Department of Labor. *Economic Report of the President*. GPO, Washington, D.C., January 1987.

Zracket, C.A. "Uncertainties in Building a Strategic Defense." *Science*, 23, 1600, March 21, 1981.