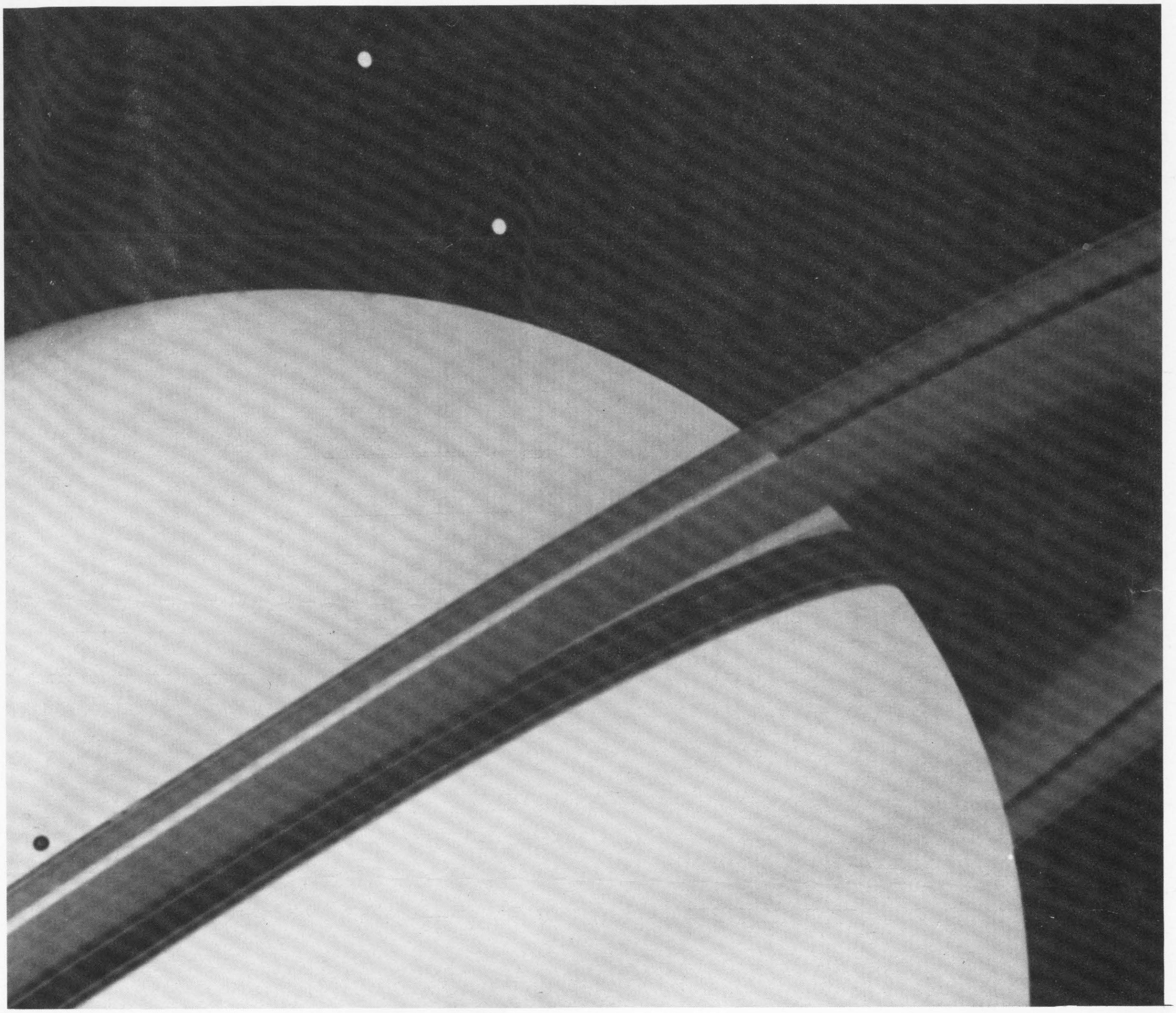
computers and people

Jan.—Feb., 1981

Vol. 30, Nos. 1-2

formerly *Computers and Automation*



SATURN WITH TWO MOONS AND THE SHADOW OF ONE – **The Flyby of Saturn by Voyager 1**

The Computer Almanac and Computer Book of Lists — Instalment 17

Neil Macdonald
Assistant Editor

66 WINNERS OF AWARDS FROM THE ACM (ASSOCIATION FOR COMPUTING MACHINERY) (List 810101)

A. M. Turing Award / contributions of a technical nature to the computing community / $2000, certificate
- A. J. Perlis
- Maurice V. Wilkes
- Richard W. Hamming
- Marvin Minsky
- J. H. Wilkinson
- John McCarthy
- E. W. Dijkstra
- Charles W. Bachman
- Donald E. Knuth
- Allen Newell
- Herbert A. Simon
- Michael O. Rabin
- Dana S. Scott
- John Backus
- Robert W. Floyd
- Kenneth E. Iverson

Grace Murray Hopper Award / outstanding young computer professional ... on the basis of a single recent major technical or service contribution / $1000, certificate
- Donald E. Knuth
- Paul H. Dirksen
- Paul H. Cress
- Lawrence Breed
- Richard Lathwell
- Roger Moore
- George N. Baird
- Allen L. Scherr
- Edward A. Shortliffe
- Raymon Kurzweil
- Stephen Wozniak

Distinguished Service Award / value and degree of service to the computing community / gift of appreciation and certificate
- Franz L. Alt
- Don J. Madden
- George E. Forsythe (posthumously)
- William Atchison
- Saul Gorn
- John W. Carr III
- Richard G. Canning
- Thomas B. Steel Jr.
- Eric A. Weiss
- Carl Hammer

Programming Systems and Languages Award / best paper in this subject published no more than 18 months before award date / $500, certificate
- Peter Lucas, Kurt Walk
- Peter J. Denning
- E. W. Dijkstra
- John C. Reynolds
- Donald R. Slutz, Richard L. Mattson
- Irving L. Traiger, Jan Gecsei
- C. A. R. Hoare
- Z. Manna, S. Ness, J. Vuillemin
- Dan Bobrow, Ben Wegbreit
- Dennis M. Ritchie, Kenneth L. Thompson
- Frances E. Allen, John Cocke
- Susan Owicki, David Gries
- Niklaus Wirth
- David Parnas

Outstanding Contribution Award / value and degree of service to ACM / certificate
- Bruce W. Van Atta
- W. Smith Dorsey
- Kathleen A. Wagner
- M. Stuart Lynn

Eckert-Mauchly Award / technical contributions to computer and digital systems architecture / $1000, certificate
- Robert S. Barton
- Maurice V. Wilkes

(Source: "ACM Awards Program" published by ACM, 1133 Ave. of the Americas, New York, NY 10036)

20 PROCEDURES THAT MIGHT BECOME GOOD PROBLEMS FOR STUDENTS OF ALGORITHMS AND PROGRAMMING (List 810102)

Hyphenating / hyphenating a word when putting part of it on one line and the rest of it on the next line
Spelling / a word, given its sounds and the sequence of sounds
Knitting / a garment, etc.
Cooking / some dish, according to a recipe
Syllables / counting the number of syllables in a word, a sentence, or a paragraph
Wrong Instructions / correcting instructions that are wrong, incomplete, ambiguous, etc.
Right Justification / right justifying a line of text, where different letters have different spacing, ranging from 2 points for "i" to 5 points for "W"
Spelling Correction / correcting the spelling of misspelled words as in "seperate" for "separate"
Spelling Correction / correcting the spelling of words that sound alike but that have different meanings, such as "there, their"
Recognition / how to recognize from facial features a person whom you have never seen
Identification / how to identify keys on a keychain according to their notches, slots, etc.
Plant Identification / how to identify a plant according to its properties

Meeting / how to meet a person at a place strange to both persons, who have never seen each other before

Four Fours / given any number N from 1 to 1000; and the operations of arithmetic, adjacency, etc.: express the number N as a function of four fours; example: 300 equals 4 to the 4th power plus 44

Arrest / how to proceed when you are arrested by a policeman

Survival / how to survive when you are lost in a wilderness probably at least 20 miles from the nearest help

Finding / how to find something which is lost, such as a baseball in a sandlot, or a paper in a dozen file drawers

Travel / how to give instructions for travel as for example to a motorist who is hunting for a street and number

Fire Prevention / how to give instructions to prevent fires in a house

Buying / how to give instructions for not purchasing useless articles or services

(Source: Neil Macdonald's notes)

7 OBSERVATIONS OF GENERAL SCIENCE (List 810103)

In any contest, there are two basic routes to victory. One is to outdo your enemy's best and wisest move. The other is to bait him into making mistakes.

"Why do you want to make an enemy like that?" -- "I need to get him involved with me, so that I can make use of him. Unless I can make him annoyed enough to thrust, I can't parry. And only by successfully continuing to parry every thrust he makes can I finally get his whole attention.

Man with a crossbow in the proper position at the proper time is worth a corps of heavy artillery half an hour late and ten miles down the the road from where it should be.

An expert is a man who knows a great deal about his subject. A scholar is someone who knows all that is available to be known about it.

There is a tactic in fencing. It is to launch a series of attacks, each inviting ripostes, so that there is a pattern of engages and disengages of your blade with your opponent's. Your purpose however is not to strike home with any of these preliminary attacks; but to carry your opponent's blade out of line with each disengage so gradually that he does not notice that you are doing it. Then, following the final engage, when his blade has been drawn completely out of line, you thrust home against an essentially unguarded man.

The complexity is more apparent than real.

Everyone is more or less the prisoner of his current situation. Manipulate the situation and the individual often hasn't much choice but to let himself be manipulated as well.

(Source: "The Tactics of Mistake" by Gordon R. Dickson, in "Analog Science Fiction", four parts, October 1970 to January 1971)

51 WORDS OF TWO OR MORE SYLLABLES DEFINED IN ONE-SYLLABLE-WORD EQUIVALENTS (List 810104)

appreciate	give great worth to
condiment	spice
confound	get mixed up
consign	put in the care of
construct	make
decadent	on the down slope
decay	go bad
echo	sound that comes back
effusive	full of words
elucidate	make clear
enactment	law put in force
encompass	put round, go round
grovel	go down on one's knees in front of
harangue	long talk
hereditary	handed down
hesitate	take a small time for thought
mercy	wish not to give pain
misconception	a wrong thought
mismanage	do (a thing) wrong
mistrust	have doubts in, not trust
misunderstand	get the wrong point
monarchy	rule by a king
monopoly	being just one in a field
moribund	being at the point of death
mutilate	wound and cut
nibble	take small bites
nocturnal	at night, by night
notorious	with a bad name
nucleus	small group
oblige	do a kind act
obstacle	thing in the way
occupy	take up, be in
occur	take place
offend	do wrong to, hurt
orate	talk well in front of men
ponder	give thought to
popular	liked by all, dear to all
precipice	straight drop down
prevaricate	say what is not true, dodge, tell a lie
prevent	keep from taking place
reprimand	say sharp words to
reprint	print once more
repugnant	not pleasant
resolute	free from fear
starboard	right side of ship
submit	give way to
subscribe	put one's name to
suppress	put an end to
tremor	shake of the earth
ubiquitous	in a lot of places
worship	go down on one's knees in front of

(Source: "The Basic Dictionary" by C.K. Ogden, published by Kegan Paul, Trench, Trubner, & Co., London, England, 1939, 106 pp, and Neil Macdonald's notes) □

Vol. 30, Nos. 1-2
Jan.-Feb., 1981

Editorial Offices	**Berkeley Enterprises, Inc.** 815 Washington St. Newtonville, MA 02160 (617) 332-5453
Advertising Contact	**The Publisher** **Berkeley Enterprises, Inc.** 815 Washington St. Newtonville, MA 02160 (617) 332-5453

"Computers and People" (ISSN 0361-1442), formerly "Computers and Automation," is published every two months at 815 Washington St., Newtonville, MA 02160, by Berkeley Enterprises, Inc. Printed in U.S.A. Second Class Postage paid at Boston, MA and additional mailing points.

Subscription rates: United States, $14.50 for one year, $28.00 for two years. Elsewhere, add $6.00 per year.

NOTE: The above rates do not include our publication, "The Computer Directory and Buyers' Guide." To receive this, please add $15.00 per year to your subscription rate in the U.S., and $18.00 elsewhere.

NOTE: No organization in Switzerland or Morocco is authorized or permitted by us to solicit subscriptions for, or receive payment for, "Computers and People" or "The Computer Directory and Buyers' Guide." Such subscriptions and payments should be sent directly to us.

Please address mail to: Berkeley Enterprises, Inc., 815 Washington St., Newtonville, MA 02160.

Postmaster: Please send all forms 3579 to Berkeley Enterprises, Inc., 815 Washington St., Newtonville, MA 02160.

Change of address: If your address changes, please send us both your new address and your old address (as it appears on the magazine address imprint), and allow three weeks for the change to be made.

computers and people

formerly *Computers and Automation*

The magazine of the design, applications, and implications of information processing systems – and the pursuit of truth in input, output, and processing, for the benefit of people.

Front Cover Picture

The front cover shows Saturn with two moons, Tethys (above) and Dione, photographed by Voyager 1, on November 3, 1980, eight million miles from the planet. The shadows of Saturn's rings and Tethys are cast onto the cloud tops of Saturn. For more information, see page 22.

Key

[A]	–	**Article**
[C]	–	**Monthly Column**
[E]	–	**Editorial**
[EN]	–	**Editorial Note**
[F]	–	**Forum**
[FC]	–	**Front Cover**
[N]	–	**Newsletter**
[R]	–	**Reference**

Notice

*D ON YOUR ADDRESS IMPRINT MEANS THAT YOUR SUBSCRIPTION INCLUDES THE COMPUTER DIRECTORY. *N MEANS THAT YOUR PRESENT SUBSCRIPTION DOES NOT INCLUDE THE COMPUTER DIRECTORY.

"THE COMPUTER DIRECTORY AND BUYERS' GUIDE" 1979-80

See announcement on page 17.

Persons, Organizations, Computers: Which of These Three Frustrates the Most?

Edmund C. Berkeley
Editor

From the dictionary:

Def.: "to frustrate"

to balk or defeat in an endeavor
to bring to nothing
to make invalid or of no effect

Synonyms: THWART, FOIL, BAFFLE, BALK, CIRCUMVENT, OUTWIT
FRUSTRATE implies making ineffectual all efforts, however vigorous or persistent
THWART suggests frustration by running counter to
FOIL implies checking or defeating so as to discourage further effort
BAFFLE suggests the interposing of obstacles or hindrances
CIRCUMVENT implies frustration by a particular stratagem
OUTWIT implies frustration by craft and cunning

Perhaps the commonest form of frustration which we the people suffer from unbridled selling and marketing in this society is "junk mail". Organizations and computers take (without permission) the precious time of people by advertising, soliciting, promoting, etc., through the mail. It takes time to look at mail, open the envelope, glance at the contents, decide about it, and if it is not interesting, chuck it out.

I have managed to look at 100 pieces of circular mail, keep about 3 of them, and throw out 97, in about 10 to 15 minutes. But it is time I have better use for.

A cable television association sends to "Computers and People" a news release; that is outside our field; I chuck it. The U.S. Department of Commerce sends C&P a news release about an exhibit of American business in Nigeria; that is outside our field; I chuck it. A big computer company sends C&P a house organ saying how Sadie Jones had a party on her 15th anniversary of being with the company; I chuck it. Unwanted or junk mail takes me at least 10 minutes day after day without my consent or approval.

Unfortunately I have to look at each piece of junk mail to avoid chucking out certain mail that I need; for example, the news of a new interesting computer application, such as the story from NASA in this issue about rowing in a row boat from California to Australia: the rowboat carries a radio beacon to inform a polar orbiting satellite of its position in the Pacific Ocean.

Junk mail may be by far the commonest form of frustration, occurring day after day for at least 100 million people. Persons produce little junk mail. Organizations produce by far the most. But computers are largely blamed for it, because the mailing labels are addressed by computers that select names from big lists according to demographic criteria. It is a kind of new power which big mailers have, and like any new power over people, the inhumanity, stupidity, and waste is thrust by those having the power, upon other people: the postal service, the buying public, and the company's treasury.

Recently I seem to have found an antidote:
1. I cross through our address.
2. I write "REFUSED" on the envelope.
3. I write "RETURN TO" with an arrow drawn to the sender's address on the envelope.
4. I obliterate any instructions by the sender to discard if not deliverable.
5. I attach a label "RETURN POSTAGE GUARANTEED", if that is not already there.

This seems to be working; I feel encouraged. Also it actually puts the burden of paying for returned mail on the mailer -- and that is a highly desirable burden, both socially and businesswise.

□

Forum

COMPUTERS AND MONOPOLY: SOME SIGNIFICANT ARTICLES

1. From David B. Sturtevant
Director of Public Communications
Association of Data Processing Service Organizations (ADAPSO)
1925 North Lynn St.
Arlington, VA 22209

I enclose a check for $26.50 to cover a year's subscription to "Computers and People" (without the Computer Directory and Buyers' Guide), two free articles:

IBM and the Maintenance of Monopoly Power, by R.M. Carlson and B. Wehrman, U.S. Dept. of Justice
Antitrust for the Computer and Communications Industry, by Senator Edward M. Kennedy

and four additional articles:

Computers, Communications, and Antitrust, by John H. Shenefield, U.S. Dept. of Justice
The Prospects of Information Tyranny, by Jerome B. Wiesner, Mass. Inst. of Techn.
Consumer Needs, and the Telecommunications Monopoly sought by AT&T, by John M. Eger, Lamb, Eastman and Keats
The Present Role of Governments in the World Computer Industry, by C. W. Spangle, Honeywell, Inc.

I am looking forward to receiving the above items.

2. From the Editor

Thank you for your letter subscribing to "Computers and People" and asking for some of the significant articles we have published in the past. Your kind of letter makes me feel happy about the role we are trying to play in the computer field. Please send me suggestions from time to time.

DJINNI CLAIMS – DISBELIEF

1. From George Sanford
1034 Chestnut St.
Chico, CA 95926

In "Computers and People" from time to time you have published information about a scheme which you refer to as "automatic computer programming using ordinary natural language" and "automatic translation by computer with due regard for meaning". The name of this scheme has been DJINNI or GENIE or DJINNI-B. You make a number of claims which are either rather or very hard to believe. You offer examples that, to say the least, are rather trivial. I believe you have been promoting this scheme at least since 1976. Yet so far I have not noticed any commercial development or marketing of it.

What are the various reasons you have found or noticed which various people give for their noninterest?

2. From Olaf Thorsen
Berkeley Enterprises, Inc.
815 Washington St.
Newtonville, MA 02160

Your question is good. Some time ago we made a list of these reasons so as to rebut them.

Technical Reasons:
- We believe that natural language programming is impossible.
- Many distinguished computer professionals have said that natural language programming is impossible, and we agree.
- The state of the art in programming declares that natural language programming is impossible; we accept that.
- Your demonstrations are incomplete. We need complete demonstrations.
- You have demonstrated only very simple examples. We need complex examples also.
- You have not demonstrated a general algorithm. We need a general algorithm demonstrated.

Business Reasons:
- The majority of my respected expert advisers say "no", and this view is binding on us.
- We have allocated our budget for this year and the next, and there is no money left that can be used for this.
- We have to concentrate our marketing on products that will be ready this year or next, and we do not believe that your scheme can be ready by then.

Desirability Reasons:
- All my colleagues and I can program easily in recognized programming languages, like BASIC, COBOL, or FORTRAN. Why bother with "ordinary natural language"?
- Almost everybody can use packaged software for almost all purposes without much difficulty. Why bother with "ordinary natural language"?

There are many facts, judgements and provable propositions which rebut these reasons. But perhaps the simplest of all is that a diskette of DJINNI is to be marketed in early 1981 by Computer Pathways Unlimited, P O Box 12892, Salem, Oregon, 97309. Then people everywhere should have an opportunity to test DJINNI and decide whether plain "ordinary natural language" such as a manager uses to a new clerk can be a satisfactory programming language, for many if not all situations. □

Data Processing Services, Competition, and Monopoly

George M. Shea
Vice President, ADAPSO, and
Vice President and General Counsel
National Data Corp.
Atlanta, GA

"All entrepreneurs operating in a competitive environment, including data processors, face a continuing struggle to retain the loyalty of their customers; to do so, they must provide superior service at competitive prices."

By way of background, my company -- like many others in the Association of Data Processing Service Organizations (ADAPSO) -- operates a nationwide system for processing information through which we offer a broad range of computer services to the public.

The Anti-competitive Impact of the Bill H.R. 6121

During the course of the last several months, ADAPSO has been actively involved in the efforts to minimize the anticompetitive impact of the bill H.R. 6121. Our concerns with this legislation are many. Before discussing the Bill, however, I would like to place ADAPSO's concerns in perspective by describing the activities of its member companies and the business environment in which they operate.

Data Processing Services

ADAPSO's members provide the public with a wide variety of computer services. ADAPSO's members account , by dollar volume, for nearly two-thirds of the computer services performed by the industry and approximately 85 percent of all remote computing. The computer services include such offerings as local batch processing (which is localized non-interactive service), software design and support (which enable users to solve problems through data processing), time sharing, and other remote access data processing services. These latter two services enable users to gain access to and make use of geographically distant computers through the use of communications provided by common carriers. Through the use of these services, it has become possible for literally anyone in the world with access to a telephone to enjoy the benefits of data processing.

Relative to the communications industry, data processing is still in its infancy. The industry is said to have been born in 1956, when the United States and International Business Machchines Corporation settled a long standing antitrust suit. Under those settlement terms, IBM was required to conduct its data processing (or service bureau) activities independent of its equipment activities through the vehicle of a maximally separated corporate subsidiary.

Based on a statement before the Subcommittee on Monopolies and Commercial Law, U.S. House of Representatives, Sept. 16, 1980, regarding the bill H.R. 6121.

Extraordinary Growth

Since that settlement, the computer services industry has experienced phenomenal growth in size, revenues, and technology. In 1966, for example, the year in which the Federal Communications Commission instituted its now famous First Computer Inquiry, the data processing industry was ten years old. At the time, a mere 700 companies were involved in the provision of computer services. Their annual revenues amounted to approximately $534 million. Today, 14 years later, over 4,000 concerns are actively involved in computer services and their revenues approach $10 billion. In international trade, these companies contribute favorably to the nation's balance of payments. At home, they have increased the productivity of the American worker.

The growth of the industry is not attributable to the expansion of one or two dominant enterprises. The data processing services marketplace is, as it always has been , a composite of large and small companies. Today, only about 50 industry participants have annual domestic revenues in excess of $25 million. Notwithstanding this fact, the United Stated has become the unquestioned world leader in computer services.

"Structural Tension" between Two Industries

Although the computer services industry has grown independently of the telephone industry, a unique relationship has developed between the two industries. This relationship is perhaps more accurately described as a structural tension. It is attributable to the dependence of data processors upon communications common carriers for the facilities needed to provide many of their computer services. (The use of communications facilities to deliver data processing services is a growing phenomenon in the computer services industry.)

Over time, this relationship has proved to be a tremendous source of innovation. All entrepreneurs operating in a competitive environment, including data processors, face a continuing struggle to retain the loyalty of their customers; to do so, they must provide superior service at competitive prices. Towards this end, data processors have devoted substantial time and resources to finding means of reducing their costs -- and thus their prices -- by making the most efficient use of the communications facilities relied upon to provide their services.

As a result of these cost saving efforts, the public has not only received service at lower prices, but more importantly, the public has benefited from the introduction of technological improvements in data communications, data equipment, and data processing. These innovations, in turn, have spurred common carriers to introduce new and improved communications services.

Innovation

I am aware that AT&T has claimed to have been a leading source of innovation over the past several decades. Without getting into a dispute as to who discovered what, let me point out that AT&T recently tried to introduce a service called "Advanced Communications Service." This offering never got off the drawing boards. AT&T simply could not make it work. Similar service, however, has been marketed for years by two, much smaller, resale common carriers. The technology used by these two carriers had its origins in the remote access data processing industry.

The varied sized of the participants in today's data processing marketplace, the needs of the diverse customers whom they serve, and the competitive atmosphere in which they operate, have all contributed to an environment in which the unique or the unusual data processing needs of a few, as well as the more commonplace needs of the many, have all been amply satisfied. Whether the need be for international access to a library of antidotes for poisons, the local batch processing of a payroll, or software solutions to business problems, the data processing requirements of the American and the worldwide public have been met efficiently, reliably, and at low cost, ADAPSO can honestly say that it is unaware of any processing needs that are currently not being satisfied by the computer services industry.

A Record of Service from Competition, not Monopoly

This record of service has been accomplished without the protection of a government-conferred monopoly. No federal regulatory agency mandated the development of today's efficient, reliable, low cost, and universally available data processing services. They are the product of an unfettered free enterprise system. Success in the computer services industry is not limited to those who possess immense resources or who control limited communications facilities. Rather, innovation, efficiency, and service to the consuming public remain the keys to prosperity. It is the pursuit of these goals which has accelerated both the evolution of computer systems and their growing importance in the daily lives of an increasing number of Americans.

I have dwelled upon the growth and structure of the computer services industry to make one simple point. The computer services industry is alive and well. The American public, American technology, and the nation's economy have all benefited from a healthy and vigorous industry. ADAPSO is therefore concerned with those provisions of H.R. 6121 which would substantially reorder the balance, as it were, in the computer services industry. ADAPSO is particularly alarmed that H.R. 6121 -- by vacating the 1956 AT&T Consent Decree without substituting an adequate replacement -- will permit AT&T to enter, and then to dominate the data processing services marketplace.

H.R. 6121 Does Not Make AT&T Compete

I want to stress that H.R. 6121 would not be as troubling if AT&T were required to enter the industry on the same footing as every other industry participant. The fact of the matter, however, is that H.R. 6121 does not require AT&T to face the test of competition.

Rather, H.R. 6121 will enable AT&T to support its competitive efforts with the monopoly power and resources generated through its control over the nation's communications network. As you are undoubtedly aware, the bill purports to prevent the transfer of such monopoly power to competitive markets by requiring AT&T to form a "fully separated subsidiary." This affiliate, however, is fully separated in name only. As a result, the structural separation required by the bill will be ineffective. It will not prevent the abuse of AT&T's monopoly power in competitive markets.

In ADAPSO's view, the only way to assure that AT&T's entry into the computer services industry is not accompanied by a dampening of competition and innovation is to require the divestiture of AT&T's competitive activities from its regulated endeavors. In the absence of such divestiture, the use of a complete and meaningful structural safeguard that functions as a surrogate for divestiture is essential. The need for such relief can best be explained by highlighting the unique communications-dependent nature of much of the computer services industry.

Dependence

Unlike many other industries, the participants in the remote access data processing marketplace are totally dependent upon one single means of delivering their product to the public. That vehicle is controlled by AT&T. No data processor can duplicate AT&T's nationwide network, nor even a substantial part of it. Building transmission facilities, even where there is room for new facilities, is out of the question for all but a handful of American corporations. Although other carriers do provide some alternatives in a number of situations, these carriers are themselves often dependent upon AT&T for the facilities they provide.

The implications of this situation are obvious. AT&T's control over an enormous monopoly customer base gives it a virtually unlimited ability to subsidize the costs of its competitive service offerings. Its control over the communications network gives it the often-overlooked ability to engage in tying. To the extent that H.R. 6121 would allow AT&T to become more than a supplier of communications service, H.R. 6121 increases AT&T's incentive to wield this power.

We live in a world today of limited communications facilities. In such a world, every circuit that AT&T provides to others diminishes -- to some extent -- the pool of circuits available for its own use. If AT&T were permitted to use communications facilities to compete with its subscribers in the provision of computer services, AT&T would have a positive incentive to provide itself with high quality service. It would have a similar incentive to provide the subscribers with whom it competes with indifferent and dilatory service.

The Power to Manipulate

This ability of the Bell System to manipulate service for anticompetitive ends cannot be overstated. Through its ability to vary the quality, availability, and maintenance of communications facilities, AT&T can reward users who subscribe to its data processing services and punish those who do not. In an industry in which the technical specifications of communications facilities assume critical importance, the ability to deliver lesser quality service is an awesome power.

ADAPSO's concerns are not academic. As two federal juries have found this year, AT&T is not above anticompetitive conduct when threatened by competition in its traditional markets. ADAPSO has no reason to believe that AT&T will not pursue a similar course in the computer services industry.

In the face of AT&T's unbroken record of anticompetitive conduct and its litany of unlawful tariffs, ADAPSO is at a loss to understand why the competitive safeguards afforded by the 1956 Consent Decree are being eliminated by H.R. 6121. It is equally puzzling why H.R. 6121 does not propose a more appropriate response to AT&T's past conduct. And, in this regard, I refer to divestiture.

The "Fully Separated" Subsidiary Is Not Fully Separated

In lieu of the Consent Decree and divestiture, H.R. 6121 proffers a singularly misnamed safeguard -- a "fully separated" subsidiary. As defined by the bill, this subsidiary will unfortunately do little to prevent AT&T from abusing its position as the supplier of limited communications facilities. The separation to which this subsidiary is subject is not only incomplete, but also riddled with exceptions. More alarming, the separation provisions of the bill endorse a structure in which AT&T and its competitive affiliate will permanently remain less than fully separated.

In ADAPSO's view, a structure in which parent and affiliate can share offices and other property, and in which they can use facilities in common is not "fully separated". Similarly, a financial structure in which AT&T can guarantee the debts of its affiliate can hardly be deemed "separate". A structure in which employees can be transferred to and fro between parent and affiliate is by no means "separate." The use of a common name, as well as a common logo, is the exact opposite of "separate." These provisions will do little to separate AT&T and its affiliate -- either in their own eyes or in the eyes of the public. Incredibly, H.R. 6121 will bless this common identification (and thus the opportunities for tying) by approving joint institutional advertising by parent and affiliate. These shortcomings are made worse by a provision that places a straitjacket on the FCC's ability to prescribe further structural safeguards.

Hardware That Can Capture Software: A Sleeping Menace

Perhaps the most serious inadequacy of the structural separation prescribed by the bill is that it will permit AT&T to continue to perform critical research, development, and manufacturing for its subsidiary. This means that AT&T's affiliate will be able to obtain components and subassemblies from Western Electric forever. Deceptively simple in name, these items have assumed increasing importance in the computer services industry, particularly as more and more intelligence -- previously captured in unbundled software -- is incorporated into hardware. As a result, H.R. 6121 will enable AT&T to use these devices as a permanent source of cross-subsidies and a vehicle through which inside information about the network can be transferred to its subsidiary.

The bill purports to limit these opportunities for abuse by requiring AT&T to sell these components to all parties. But this safeguard could prove to be meaningless. Network equipment and software can easily be engineered so that the items provided by AT&T to its affiliate will be of use to only one party -- AT&T's affiliate. Moreover, such a safeguard works at cross-purposes with structural separation. Contrary to the overriding philosophy of the bill, it will enable Western Electric, along with AT&T's affiliate, to participate directly in an unregulated competitive market on an unseparated basis.

Finally, ADAPSO is concerned that the Bill's tampering with AT&T's structure will adversely impact the course of events in the one forum in which divestiture is seriously being considered, the United States District Court for the District of Columbia. There can be no question but that H.R. 6121 will adversely affect the ability of the Department of Justice to secure the relief that it has requested. Consequently, ADAPSO does not believe that this type of legislation should be enacted at the present time.

If such a bill is to be enacted, ADAPSO urges the Subcommittee to modify those provisions of the Bill which would be the most damaging to the suit. The approach suggested by Chairman Rodino in his letter of July 21, 1980 has substantial merit. H.R. 6121 could be improved simply by striking those provisions which modify the Consent Decree and which restructure AT&T.

The American Public Does Not Need H.R. 6121 in Its Present Form

As I have indicated, the computer services industry is doing quite well. Neither we, nor the American public need H.R. 6121 in its present form. No showing has been made before the Congress, before the courts, or before the FCC that

(please turn to page 26)

Programming for Users: A Bit of Psychology

Barry Dwyer
School of Mathematics and Computer Science
South Australia Inst. of Technology
The Levels
Pooraka, South Australia 5095

"Frankly, I get along with the computer better than with most of the other people in the office."

Needs and Motivation

How can programmers make use of behavioural science to make life more bearable for people who use computers - including themselves?

Let me start with Maslow's "Hierarchy of Needs". According to Maslow's theory, human needs can be arranged in a hierarchy like Figure 1.

SELF
REALISATION

ESTEEM

SOCIAL NEEDS

SECURITY

FOOD, DRINK, WARMTH

Figure 1

When a person's needs are satisfied on the lower levels, he can seek satisfaction at the higher levels. A person whose physiological and security needs are met, will be mainly concerned with social needs; the need to make friendships and to belong to a group. Only one who has all four lower level needs satisfied, can be concerned with the highest level; the realisation of his own ideal of himself.

Union rules ensure the lower two levels are satisfied for most people in the work force. Most workers are therefore concerned with socialisation and self-esteem.

Herzberg's Theory of Motivation

A related point of view is expressed by Herzberg's theory of motivation. According to Herzberg, a need which is satisfied cannot be used as a motivator, except negatively by the threat of taking it away. If working conditions offer adequate levels of wages, comfort, and safety, improving these levels will do little to alter motivation. These he calls "maintenance factors." Motivating factors are associated with social needs and self-esteem, and for the fortunate few, the ability to pursue their long-term ambitions. A key element is the feeling of recognition or achievement.

Needs of People in Computer Programming

How do people involved in computing fill their higher needs?

For most programmers the answer is easy. They work as members of a team. This satisfies the need to socialise, and also provides "peer group recognition". Outside the peer group, a programmer finds relatively little recognition because most people have little understanding of what he does.

This is a dangerous situation. It means that a programmer is most motivated to achieve those things which win admiration from other programmers. Those are rarely the things most desired by management, or that mysterious person, the "user".

This means that management must intervene with project control systems, specifications, documentation and so on, to point the programmer's efforts in the right direction. Since such plans do little to satisfy social needs, they are ineffective unless they are "internalised" by the programming teams as group norms. This is sometimes done by letting programmers design their own standards and project control system.

Needs of Computer Operators and Data Entry Clerks

What about the other groups who deal with computers, the computer operators, and the data entry clerks? How are these people able to meet their needs for socialisation and esteem?

In the usual course of events computer operators have an uneven workload. During the slack periods there is usually plenty of time to socialise with other staff.

Recently, our installation had trouble in cutting one shift down to a single operator. The operators put forward all sorts of reasons why this was unworkable, in spite of clear evidence to the contrary. They demanded a higher duty allowance for the single shift because of the extra responsibility involved.

Basically, the problem was not too much responsibility, but loneliness.

Finally, the data entry clerk. Here is a person who spends most of the day talking not to other people, but to a computer. The only way for that person to satisfy social needs may be to take a break from his work.

Little wonder that operators of VDU's find it necessary to work 2 hours on, 1 hour off. Little wonder that they are concerned about radiation hazards or eyestrain. But in spite of numerous complaints, no one has yet found any medical evidence that VDU's are harmful.

Symptoms of Bad Programming

What can programmers or system designers do to make things better?

Here is an anecdote which illustrates the problem.

A couple of years ago I wrote an on-line enquiry/update program. In my efforts to ensure the accuracy and integrity of the data, every conceivable check was made on the input. Not only definite errors, but also potential errors, resulted in a loud "beep" and an error message.

Here are some consequences of the program design:

- The screen was too bright,
- The screen was too jittery,
- The keyboard was too high,
- The keyboard was too low,
- The chair was the wrong height,
- The beep was too loud,
- The input forms were badly designed.

I know these were consequences of the programming, because when I changed the program the problems all went away, and the data-entry clerks stopped complaining and started praising.

Further light was shed by a computer operator. I tried to make the operation of this system completely fool-proof. How did he like it? "Great" he said, without enthusiasm, "A gorilla could do it".

Systems that Are a Pleasure to Use

Since that time I have modified and improved the technique I used to correct the problem so that I can write systems which are a pleasure to use, first time.

But if you get your designs wrong, don't expect your users will tell you. They won't know what is wrong, only that something is not right. So they will complain about the feel of the keyboard, or the colour of the ceiling, or anything.

Behaviourism

The key to good systems design is an understanding of "Behaviourism". This theory says that virtually all behaviour can be explained as effects of reward and punishment. (It is also known as Skinner's Theory of Operant Conditioning).

If a certain action is usually followed by a reward, the action is encouraged. We say the behaviour is positively reinforced. If an action is followed by a punishment, it is discouraged. So much is obvious.

What is not obvious is that timing the reward is very important. More important than its intrinsic value. One word of praise now is likely to be worth more than a $10 bonus at the end of the week. Figure 2 shows the experimental relationship between the timing and effectiveness of a reward.

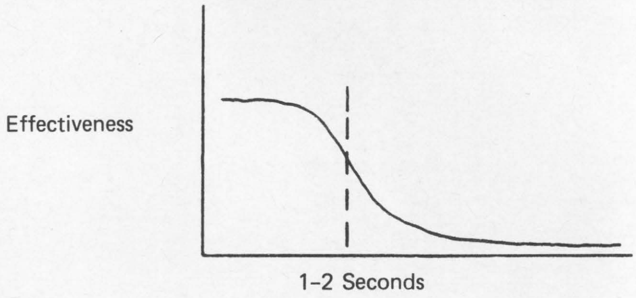


Figure 2

Another point to remember is that punishment never encourages desired behaviour. It discourages the action which was punished, but does not in any way determine what will take its place.

"I don't like it because it keeps finding my mistakes"

We can now understand the behaviour of the data entry clerks in my story. The "beep" and the error messages were punishing them in a mild but effective way. How could they avoid punishment? Keep away from the computer. How could they get away from the computer? Well any excuse would do, but they could hardly say "I don't like it because it keeps finding my mistakes."

If people will accept an error message as punishment, will they accept praise as a reward, even when the praise comes from a computer? Illogical as it may seem, they do!

Projection of Personality into the Computer

People who use a computer system soon begin to project a personality onto the computer. What sort of a personality will it be? That will depend on the man who programmed it. He is allowing certain aspects of his personality to be built into a program.

Consider the following conversation:

```
   User:  COPY, LISTING
Program:  ILLEGAL COMMAND
```

What does it tell us about the programmer and his view of the user? How does the conversation encourage the user to do the correct thing? Incidentally, "COPY, LISTING" is not an "illegal" command. It is perfectly OK in the right context. A more truthful reply would have been:

User: COPY, LISTING
Program: THE COPY COMMAND IS ONLY PERMITTED IN BATCH MODE.

This is not only less authoritarian, it also helps the user decide what to do.

Error Messages

It is commonly thought that since the purpose of error messages is to identify errors, the best error message is one that diagnoses the problem as closely as possible. Consider this:

User: CR 25.00 SAT51-2
Program: ILLEGAL ACCOUNT CODE.

This is a very confusing diagnosis. I assure you that "SAT51-2" is a perfectly good account code. The user may have to spend some time discovering that this is so, perhaps by consulting the latest chart of accounts.

Compare it with this response:

```
   User:  CR  25.00 SAT51-2
Program:  CR  25.00 SAT51-2
                --*--
          CR ACCOUNT AMOUNT
```

I want you to notice that there are three parts to the program's response, linking what the user did, to what he should have done.

Correction of Actions with Teaching and Rewards

There are some general principles involved here.

- Reward desired behaviour.
- Do not "punish" undesired behaviour, but show how it may be improved.
- Form a chain of association between the objective the user desires to accomplish, and the specific action needed to do it.
- Keep the time interval short between action and reinforcement.

Translating these principles into programs calls for some imagination. There is no "standard solution". But the following examples should give you some starting points.

Reward Desired Behaviour

All that is needed to reward desired behaviour is a simple word like "GOOD" or "OK". When a person is really familiar with a system, probably a cue character such as "?" is enough. On the other hand, if a computer typed "EXCELLENT!" every time you hit RETURN, you might begin to doubt its sincerity. Also you would regard the time taken to type EXCELLENT! as a waste. It would then become a mild punishment, what Herzberg calls a "PITA factor". I will leave you to figure out what the acronym PITA stands for.

The method I have adopted is to reward selectively. A good principle is to offer two or three positive reinforcements for every negative reinforcement. If the user has trouble with a particular command, when he gets it right, the program can say "GOOD". Not just once, but also the next couple of times he uses that format.

Random Schedule of Rewards

Better still, introduce an element of randomness into the situation. Make the probability of a reward depend on the number of errors being made. It is amazing how the user perceives this random reward pattern as carefully judged praise for work well done.

It is an established experimental result that random reinforcement is not so fast at teaching as indiscriminate reinforcement. But the results it produces are much longer lasting.

```
User:     VO 105006/SC AJAX/IN  78-105301/DB
          SAT51 2 24.50
Program:  VO 105006/SC AJAX/IN  78-105301/DB
          SAT51 2 24.50

          DB ACCOUNT AMOUNT
User:     DB SAT51-2 24.50
Program:  GOOD
User:     DB IA56-7 5.25
User:     DB IR91-1 13.
Program:  GOOD
          etc.
```

(The user will almost certainly be able to rationalize why his last response earned a reward.

Do Not Punish

From the point of view of the programmer, an error is the failure by the user to enter correct data. From the user's point of view, an error is a failure by the program to understand what he has typed. Since the dialogue takes place between the program and the user, it is as well if the programmer takes the user's point of view.

Not the Programmer's Job to Load the User with Guilt

For example instead of -
UNRECOVERABLE SYNTAX ERROR

we could have -
UNABLE TO ANALYSE YOUR COMMAND. CAN'T CONTINUE.

Instead of -
12 ERRORS IN INPUT, RUN ABORTED

how about -
297 INPUT LINES READ
285 WERE CORRECT
THE REMAINDER COULD NOT BE ANALYSED.
PLEASE RERUN.

which emphasises only the positive aspects of the user's performance. It is not the programmer's job to load the user with guilt. Another aspect of the programmer-centered error message is the use of jargon which is meaningless to the user. The programmer can easily notice this sort of thing when it happens to him. For example:

CRM ERROR 01

is of little use to a user who doesn't know what "CRM" is. How will he find out?

Deferred Correcting

In a data entry program I wrote, I had the opportunity to include on-line checking of account codes and customer codes. From my (programmer's) point of view this was a chance to eliminate errors at the earliest possible point. In fact the check was deferred until a batch of data had been prepared.

Why defer the checking? Because an error of this kind would most likely originate with the costing clerk, not the data entry clerk. So why punish the data entry clerk for doing his job? If the computer will not accept a wrong code during data entry, what is the data entry clerk to do? It is better to save these errors and deal with them in a batch. The negative reinforcement is then less effective.

A Chain of Association

There are usually considered to be two distinct approaches to computer dialogue. Either the user gives commands, or the computer asks questions and the user responds. The first is more suited to experienced users, and the second to the novice user.

An ideal situation would be to have the system adapt itself to suit the experience of the person using it. The novice would receive plenty of cues; the experienced user would not waste time with them.

Here is an experienced user talking to a program.

```
User:     VO 105015/SC AJAX/IN 78-105301/DB
          SAT51-2 24.
Program:  ?
```

Here is an inexperienced user with the same program. It is assumed he knows what the command codes are for.

```
User:     X
Program:  X
          →
          CHOOSE FROM - VO JE RV RJ FI
User:     VO
Program:  VO
          --?
          VO VOUCHER-NO.
User:     VO 105015
Program:  GOOD
User:     X
Program:  X
          →
          CHOOSE FROM - SU SC RV.
User:     SC
Program:  SC
          --?
          SC SUPPLIER CODE
User:     SC AJAX
Program:  GOOD
          etc.
```

Any Error Is a Request for Help

How does it work? Any "error " is a request for help. If the command is recognised, the correct format of the command is displayed. If the command is not recognised, a list of valid commands is displayed. The advanced user also has a facility (/) for packing a series of commands on one line.

By having a restricted amount of cue information typed, the user is constantly being encouraged to memorise formats, rather than being spoon-fed. If we had wanted to present more detail for the novice, this could have been provided at a second level of assistance.

Within a programmed reply there is again an attempt to form a chain of association.

```
VO  (This is what you typed, and it is
    wrong.)
--? (Something is missing after VO.)
VO VOUCHER-NO (The voucher number is missing.)
```

Keep the Time Short

It is not hard to keep the response time between user action and reinforcement short in an on-line environment. Even so, it is best to begin (as above) by "playing back" the user's error. This helps build the association chain. It also refreshes the user's memory of what he did, and makes it quite clear what message was actually sent and received.

In a batch system, the time between action and reinforcement is necessarily longer; too long to be effective. It is well known that learning is much more rapid in an interactive environment.

The only way to overcome the delay is to attempt to make the user "relive" the earlier experience. A play back of his input is now essential. Reinforcers should appear in context, so that reading the play back simulates the dialogue which would have taken place on-line.

This is not easy, because in batch systems the user is not able to correct his input data in response to the reinforcer. There is the danger that errors may propagate. This will suggest that good data is actually wrong.

It is important to eliminate spurious error messages. Apart from being confusing, they negatively reinforce correct behaviour, and lower the value the user will attach to genuine error messages.

It is best to skip input data until a clean start can be made, even if this means that some real errors will be missed.

A Helpful Person Volunteers Help

Considering the behavioural aspects of computer dialogue will cause very little extra work for the programmer. But it will result in big changes in user attitudes and motivation.

The user perceives the computer as having a personality. If that personality is friendly and helpful he will enjoy interacting with it. There is nothing weird about this. The user is not talking to a computer. He is talking to another person, the programmer. The program-

(please turn to page 26)

The Technique of Software Automation

R. W. Kristinat
Editorial Director
"Vectors"
Hughes Aircraft Co.
Bldg. 100, M/S C680
Box 90515
Los Angeles, CA 90009

"A computer's tenacity for consistency makes it ideal to enforce engineering discipline during the design of software."

Based on an article in "Vectors" Summer 1980, and reprinted with permission.

Genetic engineers are learning to manipulate the formation of living organisms that produce an array of beneficial substances including human insulin, growth hormones and bacteria that literally feed on oil spills. New life forms created in laboratories by gene splicing, carefully restructuring an organism's genetic code, soon will become extremely productive tools to enhance the quality of life for humans.

There's another sort of "genetic engineering," also in early stages of technological development. And it too promises to benefit humans with a more productive, albeit inanimate, tool. Where gene splicing controls the "blueprint" of a new organism's developmental characteristics, software engineering controls the functional characteristics of a machine—probably the most productive machine ever invented: the computer.

Computer capacities have proliferated because of their microelectronic circuitry—the hardware that can receive, process, store and transfer immense volumes of information faster than the eye can blink, more consistently than the brain can comprehend.

Yet despite its speed and capacity, this machine still needs to be told what to do, how to do it and when. A computer needs direction from people before it can flex its microelectronic muscle.

Direction is provided through software which guides and organizes hardware functions. In essence, software is the machine's genetic blueprint, much as DNA* is a living creature's genetic blueprint. It determines a computer's operating characteristics.

The process of software development defines how the hardware will function and how the functions will be organized in hierarchical fashion. Every significant contingency must be dealt with, every step in the computing process precisely described. Then the intricate sets of instructions, or modules, must be translated into language a machine can understand.

**DNA, deoxyribonucleic acid, is the molecular basis of heredity carried by genes.*

as computer tasks get more complex, the complexity of software development mounts by several orders of magnitude.

In earlier days, when tasks were relatively simple, software development was considered a craft quite apart from the engineering sciences applied to hardware development. The quality of a software program hinged on the skill and ingenuity of an individual artisan.

As hardware advancements offered greater capability in smaller packages, innovative users began heaping heavier, more complicated workloads on computers to fully exploit the potential. This was particularly true in military and aerospace applications where esoteric, one-of-a-kind systems were required for super-critical tasks.

The problem is that as computer tasks get more complex, the complexity of software development mounts by several orders of magnitude. Some examples:

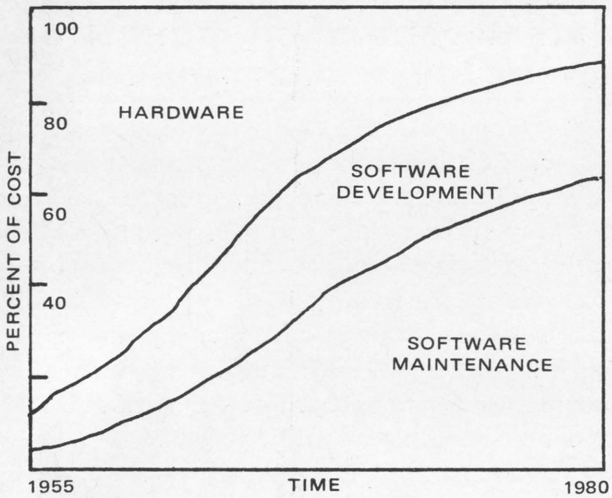
■ A typical software program in the mid-'50s might have had 10 operating modules and used 1/100th the computer capacity used today. By comparison, it is not unusual to see 12,000 software modules integrated by a staff of 100 software specialists in modern aerospace and military computer systems. As many as 10 different computers may interact with one another.

■ A single computer function today could require 200,000 machine instructions, and there could be 25 separate functions involving the guidance of a geosynchronous satellite from earth launch to its assigned orbital slot 22,300 miles overhead.

■ An advanced weapon system incorporating radar and imaging infrared sensors with command, control and communication networks might require up to 500 man-years of software development and maintenance over its lifetime, using conventional procedures.

The growing complexity and scale of computer systems placed an enormous strain on software in terms of resources, costs and reliability.

Industry and government authorities predict a ten-fold increase in automation during this decade — and a 30 percent shortfall of trained software specialists by the mid-'80s. It's a labor-intensive process. People not only originate the ideas that form the initial blueprint of characteristics. When user requirements change or technology improves, people have to modify existing software without violating the integrity of the whole system. This so-called lifetime maintenance is draining off the labor pool as more software systems go on-stream.



Software Cost Trend

Not surprising, software costs accelerated. In the mid-'50s, software accounted for about 15 percent of all computer costs; hardware, about 85 percent. This year, a quarter of a century later, the percentages are reversed!

Moreover, software reliabilty began falling short. Computer users stretched their requirements to take advantage of hardware capability, and sometimes they were disappointed. Software could not always deliver on the promises of the computer's hardware.

As a result, software was tagged with an oft-deserved reputation: over-priced, over schedule and under-performing.

Clearly, changes were in order if software productivity and reliability were to keep pace with the rapid progress of hardware technology. Software development, being fragmented by an avalanche of complex and large-scale computer missions, needed something to hold together the increasingly splintered process of formulating machines' genetic blueprint.

software is gradually but inexorably making the transition from craft to technology...

That something is the rigorous discipline of software engineering, the genetic engineering of machines. Software is gradually but inexorably making the transition from craft to technology — a technology with standards and practices more demanding than those of hardware.

As hardware engineers had done to cope with miniaturization, the new breed of software engineers is turning to computer-aided design techniques to help raise their productivity and their system's reliability. Computers were used for translating programmer instructions into machine languages, relieving the drudgery of manually transferring a program concept into higher order language and thence to binary codes that machines can understand. The true potential of computers, however, is being utilized most fully during early stages of concept and design of software. In those early stages, software teams come to grips with the fundamental issues.

Boiled down, any software process is basically problem solving: identify the problem; then select the most appropriate measures that lead to a solution. If software originates from misconceptions about a mission, the computer system's performance will never match its requirements. If elements violating a system's innate nature are inflicted on its genetic blueprint — whether it be a machine or a living organism — the system will degrade or even fall apart.

A major stride aimed at avoiding these hazards is an automated requirements analysis system being used in manufacturing, management and computer sciences at Hughes Aircraft Company.

boiled down, any software process is basically problem solving...identify the problem, then select the most appropriate solution...

First, the system verifies that requirements are clearly, completely and concisely spelled out before software design is initiated. The computer flags terminology that may be subject to misinterpretation in user specifications. Hazy generalities are reduced to very objective specifics so that user and software team alike agree on the full dimensions of the problem.

Once requirements are clarified, a software designer can use the computer to help validate an idea and compare it with alternative approaches. With push-button ease, the designer sits behind a TV-like terminal and watches the idea flower or flop as the computer simulates the software concept's function before a single line of programming is generated. Throughout this simulation analysis, the designer and machine communicate via the terminal in software design vocabulary.

Requirement and design concept verification is only the first step in automated software engineering. The rigorous discipline needs to be carried throughout the software process.

'it won't make a genius out of a dolt, but it'll keep the dolt from ruining the design and free the genius for more creativity..'

Hughes is developing Automated Interactive Design and Evaluation System, AIDES for short, to increase productivity and reliability by enforcing standard procedures throughout design of software. A computer's tenacity for consistency makes it ideal to enforce engineering discipline during the process. For instance, fuzzy thinking simply "will not compute" because only sound engineering principles reside in the data bank. Furthermore, errors or omissions are detected before locked in design.

As one software engineer puts it: "AIDES won't make a genius out of a dolt, but it'll keep the dolt from ruining a design. And it'll free the genius for more creativity."

A proposed design can be measured against alternative possibilities in terms of quality, using as criteria a design's simplicity, economy, adaptability and testability. The designer will use the computer terminal like an automated "engineering scratch pad" to evaluate structures and architectures that lead to a solution of the problem. Periodically, the designer will be able to figuratively stand back and see how the particular subsystem fits into the total scheme, even without knowing details of an entire complex system.

Besides serving as a high speed information source for design decision-making, AIDES also will automatically draft, store and document the latest software design. The AIDES computer will draw and position all software components. It will rearrange structures when changes are made and generate wall-size structure charts, visual representations of the genetic blueprint. AIDES is expected to save 95 percent of all structure chart costs as well as trimming the software design time by about 30 percent, when implemented next year.

There's another perhaps more far-reaching benefit from the computer-aided system. Once a software module design is completed, it is filed in the system's data base. This expanding library ultimately can serve as a storehouse for easily accessible off-the-shelf software. A computer's "genetic engineers" will be able to reach into the vast inventory and draw upon reusable software modules just as their counterparts, hardware engineers, pluck the microelectronic building blocks off the shelf.

Before that, however, automated software engineering needs to be extended to other software phases: programming, testing and maintenance.

But at least the pathway leading to automated software engineering has been found by this new breed of genetic engineers for machines. Using the inherent capabilities of the machines they help design to solve problems for others, software engineers are beginning to solve their own software problems. □

NOTICE

To Our *D Subscribers:

The Computer Directory and Buyers' Guide, 1979-80

is again late, we are sorry to report.

Again, we tried to produce it by a computer system, which ought to be a much better procedure. That has not worked.

We have now decided to produce it once more by clerical typing, in the old way.

We hope it will go to the printer in early 1981.

Edmund C. Berkeley
Editor

The Impact of Automation on People — Part 2

by a Panel of Eight Individuals under the auspices of the National Bureau of Standards, Washington, DC, consisting of

1. *Wil Lepkowski, Science and Technology Editor, Washington News Bureau, McGraw Hill World News*
2. *Ben Bova, Editor, Analog Magazine*
3. *Dr. John McCarthy, Director, Artificial Intelligence Lab., Stanford Univ.*
4. *James Albus, Project Manager, Office of Developmental Automation and Control Technology*
5. *Daniel V. DeSimone, Deputy Director, Office of Technology Assessment*
6. *L. K. O'Leary, Asst. Vice Pres., Amer. Tel. and Tel.*
7. *Dr. Michael MacCoby, Psychologist*
8. *Gus Tyler, Asst. Pres., International Ladies Garment Workers Union*

"For progress there is no cure. Any attempt to find automatically safe channels for the present explosive variety of progress must lead to frustration. The only safety possible is relative, and it lies in an intelligent exercise of day-to-day judgment."

Based on a Symposium at the National Bureau of Standards in 1974.

7. Who Creates the Technology of Automation and Why?

by Dr. Michael MacCoby, Psychologist

Whenever there are discussions of technology and the future, there is a great danger of getting into a very abstract point of view, a point of view which does not really take into account either real people or real social and political factors. One gets into the situation of talking about all kinds of possibilities. In fact, if one goes to any of the major corporations that deal with advanced technology, one finds probably hundreds, if not thousands, of extremely brilliant, creative people who have all kinds of ideas, most of which never become produced or developed.

The question that one should start with is: Who creates the new technology and why? And in looking at this question over the past decade, I think that one can generalize, even though it's always risky to do so, that the major thrust of technological development, of large scale technological development, has obeyed two kinds of principles.

One is technology developed by the State, where the underlying two-fold principle has been: (1) an increase of security, both internal and external security, and (2) the principle of glory and prestige. This incidentally has had a lot to do in the United States and the Soviet Union with the space program.

Now those principles, both of security and prestige, from a psychological point of view, very easily lead to deep irrationalities, because if you are a little bit paranoid and you have some reason to be so, since there are always perceived threats, there is never any limit to the amount of security that is secure. And, equally, if you are grandiose, there is never any limit to the amount of prestige or glory that you can have, as witness the desire to have the moon, which has been achieved by one of our recent administrations.

The second principle is the principle of corporate growth and profit. Now the principle of corporate growth and profit is less irrational than that of the Government, because at least it must obey some kinds of principles of market, what sells, and furthermore, in terms of production process, what people are willing to do. Because you can have any kind of producion technology and if people are neither educated nor willing to do it, it can't be done. Often there is a symbiosis between the Government principle and the principles of the corporation. Sometimes there is not a symbiosis, but rather a conflict instead, when the Government's irrationality threatens good business practices, management, and planning. And also interferes with development of markets. This is happening, but I won't go into it now. Now I wish to make two points.

The first is to stress that the motives for developing new technology, from the assembly line to the computer, have not been to further the humane development of individuals who work with it. While some technological development may have humane consequences, such as doing away with bad jobs or increasing communication, most managers who create new technology never even think of the human consequences of what they are creating. I don't say this theoretically. I say this after a group of my colleagues and I have traveled across the country interviewing the leading creators of new technology in corporations. They are neither inclined by character or more educated by institutions to consider how technology affects emotional health and ethical development. Most corporate managers never consider the relation of new technology to: love of life versus mechanized attitudes; psychological activeness versus passive consumerism; and greater social equities versus inequity and resentment.

Managers do generally believe they are benefiting humanity by (1) creating jobs, (2) making life easier for people, (3) increasing communication, and (4) generally raising the standard of living. But even in cases where they believe they are helping people by creating new technology, social goals do not determine the priorities of what is produced. The priority is always what can be sold the most easily or what is demanded by the State.

Some more progressive managers do say that the modern technology of production has, in fact, been dehumanizing; it has instituted systems of rigid hierarchy and fragmentation of work. But they do not relate these to their social-psychological effects: lack of responsible citizenship from those who have no say in the work place; deadening of sensibility due to loss of craftsmanship; loss of the power to determine the pace of work in human life according to individual human preferences and choices.

I am speaking as a social, psychoanalytic commentator; and I must say sadly, that given the current system and the current principles that determine technological development, it is a very great likelihood that in the future we shall move towards a society of two classes.

One is the class of technocrats who, psychologically, are people who are detached, highly competitive, game players, who are all head and little heart, who live in highly protected and privileged enclaves.

The other class is the masses, the class of people who "have not been able to make the grade", the losers, people full of resentment, people who are expected at best to be consumers, but who at worst are empty and passive. They look for excitement, drugs, more and more violence, to "turn them on".

This development is all around us in the United States today. There are many signs of it that we can read every day in the newspaper.

Consequently, we have to look at the impact of automation on people in a totally different way.

8. 25 Years of Transformation of the United States by Automation

by Gus Tyler, Assistant President, International Ladies Garment Workers Union

Automation has had a most profound impact on American society, especially in the period from the end of World War II up to the present.

When we were coming out of the second World War, the assumption was that because of automation, in part, we would be moving into a period of massive unemployment. Prior to World War II, there had been technological advances, but the second World War speeded it all up. The nation was given an almost impossible challenge: how do you remove 12 to 14 million people from the civilian labor force and then meet the demands of both peacetime and wartime production, which is like doubling it all? That was the spark to step up productivity; we came out of the War with a thing called automation, which was old stuff with a new name on it.

It was assumed that with these smart machines displacing men, you would have growing unemployment. With 12 million people or 14 million returning from the war, you would have still more unemployment, and labor said; Here it comes, back to 1929. But 1929 never came back. From the end of World War II right down to the present, the size of the labor force has grown. The participation rate, that's the percentage rate of people over 14 in labor, has grown. Unemployment has shrunk; and my guess is that it would be at the present time no more than 3 per cent if Nixon had not been elected President of the United States. But even with him, it only goes to 6 per cent; so it never happened.

Why did it not happen?

For several reasons. The workers who had been working 44 and 48 hours a week during the war and were getting overtime went on strike. They said they wanted the same pay for 40 hours, and they got it. That expanded buying power.

Secondly, people had been on a program of forced savings; they now began to spend their money; that expanded buying power. Thirdly, unions had been negotiating all kinds of pension and welfare funds and now they were spending the accumulated money; that was buying power. Fourth, Government programs -- Social Security, etc. -- were maturing, moving onto the market; that was buying power. Fifth, new families were being formed, an optimistic moment, and they bought on the installment plan if they had good credit; that was buying power. Sixth, The Federal Government instituted a program that was called the GI Bill. It spent billions of dollars on education, on financing new business and buying homes; that was buying power. Then we had some foreign aid programs in which the United States took its money and sent it to the other countries so they could buy from us; that was buying power.

The result was that when we were all done, 12 million came back -- they got jobs, they used the automated equipment, they expanded our production, and we had no unemployment, because with increase of supply we had simultaneously increased demand and we were able to move forward as a Nation.

Point No. 2. The unexpected did happen: the composition of the American labor force did change basically. Automation had its first impact really in the manufacture of commodities, both in the factory and in the farm. Hence, a smaller percentage of the work force was able to produce the necessary commodities for the entire society.

But, automation was not able to distribute those commodities. Therefore, there was a tremendous expansion of people employed in advertising, merchandising, communications, packaging, transporting, wholesaling and retailing. That's part of the service economy. Secondly, with the rise in affluence, the tastes of people changed. They didn't abandon their desire for commodities, but they expanded their desire for services. A hair-do, the beauty parlor once a week; home repair; radio repair; car repair; hotels, motels, restaurants; leisure time activities. So we had a tremendous expansion in that service sector of the economy.

No. 3: We had grown accustomed to a thing called the Welfare State. After World War II, the rapid expansion of the Welfare State was no longer at the Federal level, but also in State, county and municipal employment and in the educational system, so we had that change in the labor force. Result: By about 1950, we had a fundamental change in the composition of American labor. Previously, most had worn blue collars. Now, most wore white collars. Previously, most workers manufactured commodities. Afterwards, most were engaged in dispensing services. Previously, the great creator of new jobs was the private sector; afterwards, the great creator of new jobs was the non-profit, primarily the public, sector. Finally, there's no longer a great expansion at the semi-skilled and unskilled level, but at the skilled level. The professional employee began to grow more rapidly than any other occupational category in the United States.

This was a new kind of a labor force. The ratios changed. Manufacture continued as big as ever. Blue collar as big as ever, in fact bigger than ever, but proportionately less; and so it changed the composition of the American labor force, and of the American labor movement, so that since the mid-1960's the unions that grew most rapidly were the new unions -- The American Federation of State , County and Municipal Employees; Government employees, educators, people in the service sector. These people are more political than the people who preceded them, and that's what changed the character of the American labor movement; it is now more political than it has ever been in its entire history.

The new kind of economy created a demand for women and also made it possible for more women to enter the labor force. At the present time, about 40 percent of the labor force is female. This is largely due to the fact that we have an automated economy. Automated economy produces a tremendous expansion in the service sector. Much of the service sector can use part-time employment; so a woman can take care of her home part of the day, and also work part of the day. Also, it is work scattered in the neighborhoods -- small units -- you work in a nursery school, you work in a doctor's office, a dentist's office, as a school aide, and so on and so forth. That is female employment. Also in manufacture, much of the work has been simplified and made lighter, and women go to work in factories.

There is a second reason why women are able to move so rapidly into the labor force (and soon there will be a female majority in the labor force). This is due to the fact that the home as an economic unit has been increasingly automated. There is now a washing machine and a dryer; there's a baby diaper service and you don't have to prepare the foods -- the foods come frozen or the foods come in cans. The result is the home has been changed. This has a tremendous impact on the new generation that's coming up, about ten million kids under the age of 16, many of them under 6 and under 3, who have both a father and mother, and who grow up parentless because both of them work. This is bound to have an impact upon the child who grows up without a parent on the spot, and it's going to bring tremendous pressure to bear on the society in terms of providing some means of raising these children -- comprehensive child care and the rest -- and the political implications are endless.

Also the implications on sex roles are endless on the nature of the home and the nature of the child. All of these are involved as the woman becomes increasingly the dominant earner in the United States. May I say in passing, that if women were to withdraw from the labor force at the present time, the number of families with father and mother who live in poverty, now 4 per cent, would rise to 13 per cent, and that if the women were to withdraw from the labor force at this time, those middle class families that buy homes could not pay the mortgage, and the middle class families that send their kids to colleges could not send their kids to colleges. The women are the big guns in the war against poverty, in the buying of homes in America, and in the education of the middle class of this country.

Final point -- automation has had a tremendous impact on American farming. I don't want to call it automation, but it is. It's the application of scientific techniques -- whatever you may call those things -- on creating the green revolution. We now grow eight stalks where we grew one before. End result: about one million people per annum have found themselves obsolete and in excess in rural America. Most of them came off the farms, other people were dependent upon their income. It wasn't simply that we were learning to produce more -- we were also pursuing an economic policy that tried to curtail production in order to maintain fairly high farm prices. When you add the two, we got a resultant wave of migration out of rural America into urban America. This happened at a most unfortunate time; urban America was already over-crowded. Much of metropolitan America's central cities were suffering with decaying and badly organized plants, and were living in financial debt. Into this mess came one million people per annum, and over about 22 years about 22 million people. They were totally unprepared for urban life, unacculturated, not prepared for jobs. The educational system in the cities was not able to accommodate them. Housing was not there for them, and so there arose in the American cities several million people that could only be described as "urban nomads." Hence, crime, disorders, riot and chaos in urban America -- one of the impacts of automation on people in the United States.

This is the period, from 1945 to the present, 1974 as I read it. I don't want to go into extrapolations at the present time -- because the moderator says "Your time is up." □

Computers: Some Forecasts, 1949, 1980

Edmund C. Berkeley
Berkeley Enterprises, Inc.
815 Washington St.
Newtonville, MA 02160

"Every defined intellectual operation will be performed by computer faster, better, and more reliably than by a human being."

Based on an invited article in "The Actuary: Newsletter of the Society of Actuaries", November, 1980, published at 208 So. Lasalle St., Chicago, IL 60604.

In 1949 a book of mine "Giant Brains or Machines That Think" was published by John Wiley and Sons, New York; in that book Chapter 11 was entitled "The Future: Machines that Think, and What They Might Do for Men". That chapter contained 12 forecasts.

What has happened? How do those forecasts stack up after 30 years?

It has been a delight to me to be surprised many times over, 1950 to 1980, as the computer field appears on its way to becoming the most important industry in the world. Why? Because it solves great numbers of problems far better than the mind and muscles of man.

The Pen and the Sword

Chapter 11 in "Giant Brains" began:

"The pen is mightier than the sword, it is often said; and if this is true, then the pen with a motor may be mightier than the sword with a motor.

"In the Middle Ages there were few kinds of weapons, and it was easy for a man to protect himself against most of them by wearing armor. As gunpowder came into use, a man could no longer carry the weight of armor that would protect him, and so armor was given up. But in 1917 armor equipped with a motor and carrying the man and his weapons came back into service as the tank.

"In the Middle Ages there were few books, and it was easy for a man to handle nearly all the information that was in books. As the printing press came into use, man's brain could no longer handle all recorded information and the effort to do so was given up. But in 1944 a brain to handle information, equipped with a motor and supporting the man and his reasoning came into existence -- as the sequence-controlled calculator."

At the end of that chapter appears:

"We can even imagine what new machinery for handling information may someday become: a small pocket instrument that we carry around with us, talking to it whenever we need to, and either storing information in it or receiving information from it. Thus the brain with a motor will guide and advise the man just as the armor with a motor carries and protects him."

Silicon Chemistry vs. Carbon Chemistry

Naturally I did not foresee chips of silicon (or germanium) on which 64,000 computer circuits could be imprinted: very large scale integrated circuits (VLSI). This is on the order of an entire computer in the space of a quarter of a postage stamp.

But a human brain with its biochemical construction is able to store probably close to 10^{11} (100 billion) bits of information. The silicon chip (or some other device) still has a way to go -- but I am sure that it will go there -- and beyond, up to the relativity limits.

Predicted Devices

That chapter forecast a number of devices; see Table 1.

Table 1

PREDICTED DEVICES AND PRESENT STATUS

No.	Device	Present Status
1	Automatic Address Book	Done. New name: automated mailing lists
2.	Automatic Library	Done
3.	Automatic Translator	Done. World Translation Co. of Canada, for example
4.	Automatic Typist	Largely done. New name, word processor
5.	Automatic Stenographer	Beginning. Some companies claim distinguishing 250 spoken words
6.	Automatic Recognizer	Several elements done, but not most
7.	Automatic Controls	Done
8.	Weather Brain	Not yet done
9.	Psychological Testing	Done. New names, automatic diagnosis, drill,...

(please turn to page 26)

The Flyby of Saturn by Voyager 1

Neil Macdonald
Assistant Editor

"We have learned more about the Saturn system in the past week than in the entire span of recorded history."

Based on a series of releases and photographs issued by NASA.

Friday, Nov. 7, 1980, 12:30 pm, Pac.Std.Time

Photographs of Saturn taken by the planetary probe Voyager 1, show many Jupiter-like features in the atmosphere of Saturn. The probe is now 4.4 million miles from Saturn.

There is a red spot in the southern cloud bands of Saturn. It is oval and roughly about 7 miles from one end to the other. There are about 2 dozen zonal bands in Saturn's southern hemisphere; and 95 individual rings inside Saturn's bigger rings.

Saturday, Nov. 8, 12:30 pm, PST

Another moon has been discovered, now that Voyager 1 is 3.6 million miles to flyby. The moon is about 50 miles in diameter, and orbits Saturn about 500 miles outside the A ring, the planet's outermost large ring.

Sunday Nov. 9, 8:30 am, PST

Six of Saturn's 14 known moons are being photographed today. The spacecraft is now about 2.9 million miles from Saturn, and speeding towards the planet at 35,000 miles per hour.

Monday, Nov. 10, 8:30 am, PST

The photographs streaming to Earth (950 million miles away) from Voyager 1 show vague surface features on several of Saturn's small icy moons, and show great detail in Saturn's cloud bands. Voyager 1 is now about 2.1 million miles from Saturn.

Tuesday, Nov. 11, 12:00 noon. PST

The photos show a gas atmosphere on the moon Titan, the largest moon in the solar system, and a dark north polar cap of thicker haze on Titan. Voyager 1 is now about 1/2 million miles from Titan, and will pass it at 9:40 pm tonight. At that time it will be about 2500 miles above the surface of Titan. The small icy moons, Rhea, Tethys, Dione, Enceladus, and Mimas are being scrutinized today.

Tuesday, Nov. 11, 11:00 pm, PST

Photos taken shortly before the closest point to Titan showed an almost featureless haze, hiding the moon's surface from view.

The closest approach to Saturn will be at 3:40 pm PST tomorrow, and the speed now of the space ship is about 38,000 miles per hour.

Wednesday, Nov. 12, 1:00 pm, PST

Pictures of Saturn's rings taken by Voyager 1 are bringing great surprises. Two eccentric rings were discovered yesterday. And there are two bright components of the F ring; they appear to be braided. Puzzling geological formations are coming into view on the small moons, Mimas, Dione, and Rhea.

The flyby will be at 3:46 pm, and will occur 77,000 miles above the planet's hazy atmosphere. This evening Voyager 1 will angle up through the plane of Saturn's rings, and start its long journey out of the solar system to the stars.

Some Saturn Data

The ancients believed that Saturn was the most distant planet from the Sun. Not until Sir William Herschel discovered Uranus in 1781 did anyone know of the existence of planets beyond Saturn.

Saturn is the only planet lighter than water: its density is 0.7 grams per cubic centimeter. Saturn was believed to be the only planet encircled by rings; Uranus' rings were discovered in 1977 and Jupiter's in 1979. The Dutch astronomer Christian Huygens correctly identified Saturn's rings in 1655.

Saturn's volume is 815 times Earth, but its mass is only 95 times. Saturn's equatorial radius is 37,500 miles but its polar radius is 33,500 miles, a consequence of swift rotation in 10 hours 40 minutes. But the Saturn year is 29.5 Earth years. Saturn receives only about 1/100 as much sunlight as that which reaches Earth.

Saturn's core of iron and rock extending out 8600 miles from the center is so compressed that it contains 15 to 20 times the mass of Earth. Surrounding the core is a layer of electrically conductive metallic hydrogen under immense pressure. The outermost layer of Saturn is an envelope of hydrogen and helium. The velocity of winds at Saturn's cloud tops is apparently 900 miles per hour.

Computers

As has been said previously in these pages, there are three "science-engineerings" which make possible travel in space: rocket propulsion; electronic communication; and computers. Computerized navigation in space and computerized imaging help produce these marvels of 1980, whereby Dr. Bradford Smith, chief television scientist at the Jet Propulsion Laboratory, can say "we have learned more about the Saturn system in the past week than in the entire span of recorded history." □

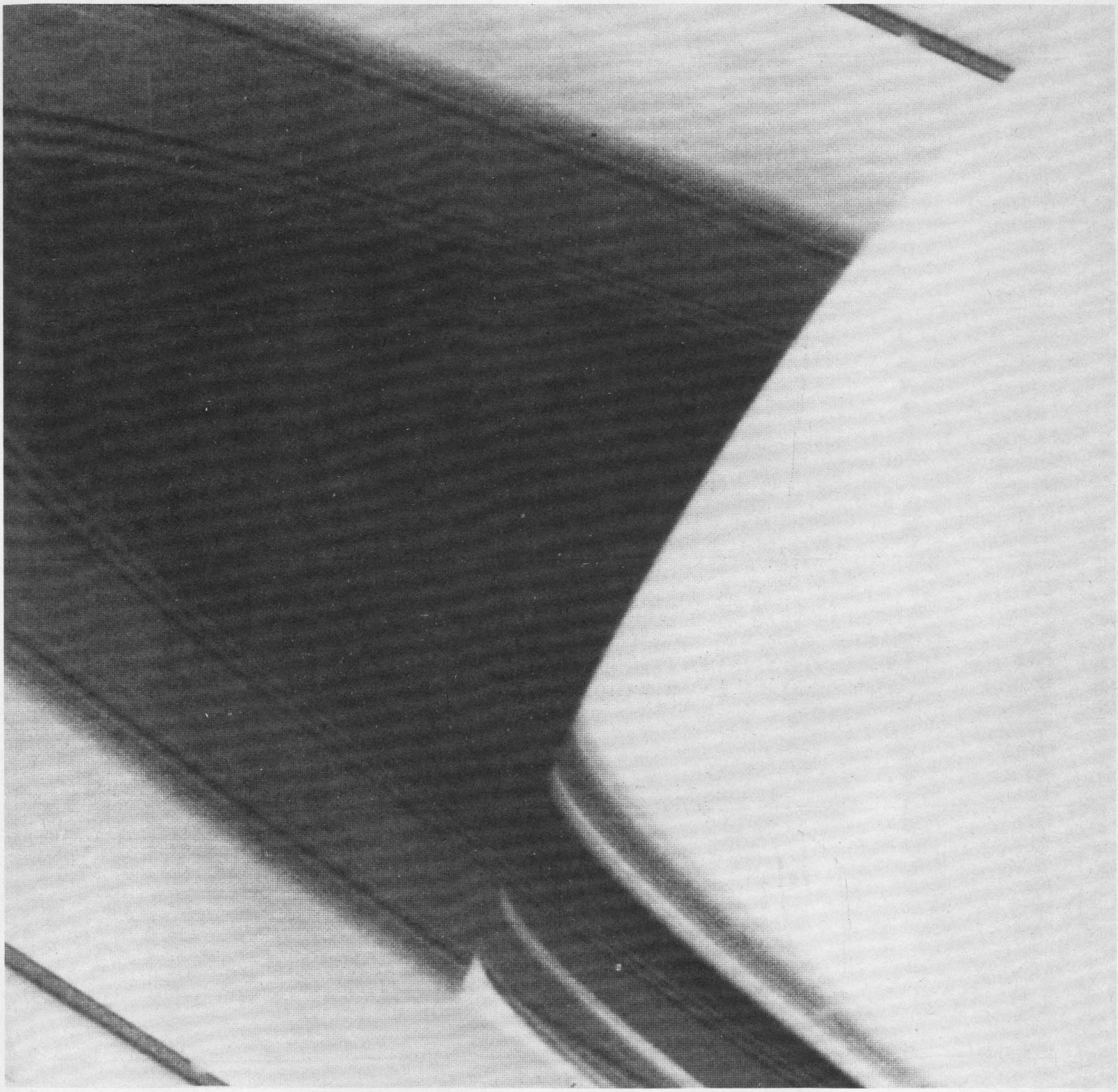


Figure 1 – This picture was taken by Voyager 1 on November 9 at a distance of 3 million miles from Saturn. On the surface of Saturn appear shadows of the rings, and shadows of ring regions of high transparency.

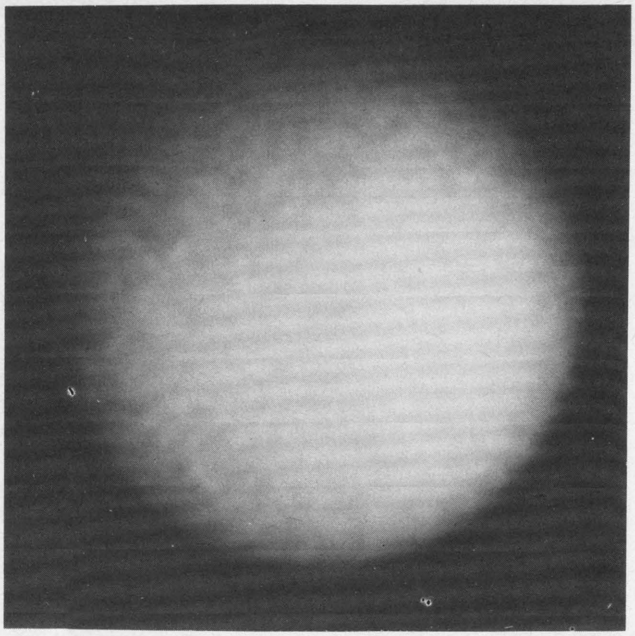


Figure 2 – Voyager 1 took this picture of Titan, the largest one of Saturn's 14 known satellites, on Nov. 4, 1980, at a distance of 7.6 million miles from the planet. The color recorded was hazy orange; it is believed to be produced by hydrocarbons in Titan's atmosphere. The surface of Titan was not seen.

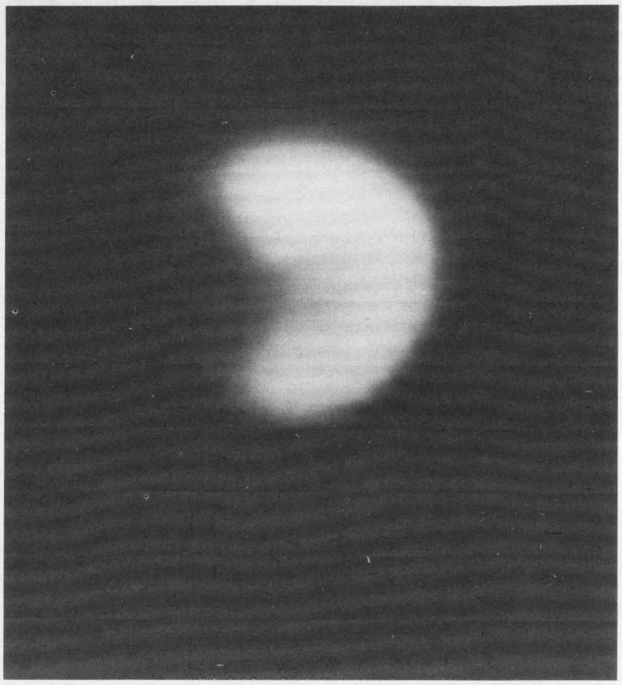


Figure 3 – This picture of Iapetus, satellite of Saturn, was taken on Nov. 6, at a distance of 5 million miles. The leading side of Iapetus is dark; the trailing side is 5 to 6 times brighter. Iapetus was discovered by Gian Domenico Cassini in 1671. It is thought that the different refelectances are due to the presence or absence of ice frost.

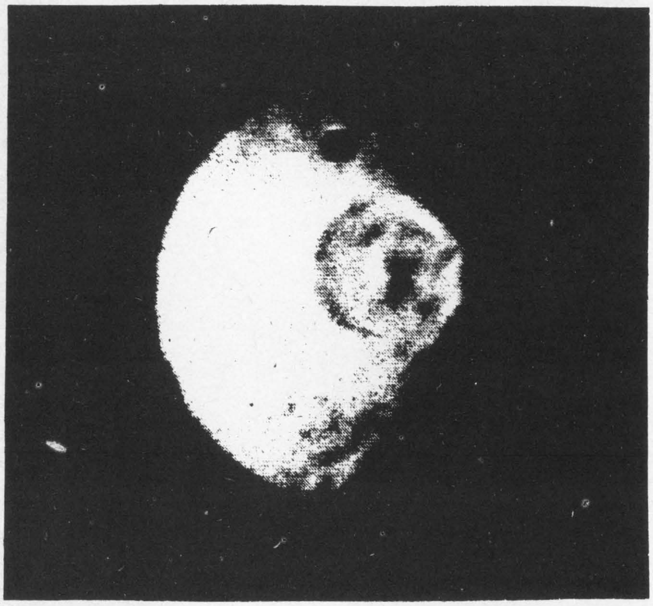


Figure 4 – This picture of Mimas, satellite of Saturn, was taken on Nov. 12, at a distance of 400,000 miles. The conspicuous impact crater with its central peak is about 80 miles in diameter. This satellite has a low density, and its chief component may be ice.

Figure 5 – This picture of Dione, satellite of Saturn, was taken on Nov. 12 (?) at a distance of 500,000 miles. Bright, wispy streaks may be frost. Craters are visible. Dione is about 700 miles in diameter.

Computing and Data Processing Newsletter

A DATA BASE INQUIRY SYSTEM WHICH SHOULD SAVE THE DEPARTMENT OF DEFENSE OVER $200 MILLION PER YEAR: LOUIS

Staff Sergeant Victoria Raun
HQ Air Force Communications Command
Public Affairs Office
Scott Air Force Base, IL 62225
618-256-4396

Louis C. Elsen claims he has "a simple mind". Elsen, a civilian employee with the 2199th Computer Services Squadron, has developed a unique computer system that is projected to save the Department of Defense more than $238 million during the first year it is used.

The new system, entitled "Logical On-Line Users Inquiry System" (or "LOUIS" for short) has been accepted as one of the Air Force's standard inquiry systems.

Elsen says the system is the result of four years of work. He says that,"Any person can in plain simple English tell the computer what he or she wants it to do -- just as if the user were talking to a computer programmer." A person not familiar with computer programming requires only about five minutes instruction under LOUIS to get information from a computer data base. "Information on personnel, budgets, finance, inventories, status of equipment, availability of communications circuits, etc., can be obtained easily and without paperwork." In AFCC computer inquiries per month have increased 500 percent in the last six months, but the total time required to answer the queries has been reduced. "Timing studies have shown that it is faster to use LOUIS than to try to read through a large computer listing."

Elsen has no formal education in automatic data processing, and claims he started working with computers "on a bet." "When I was on active duty with the Air Force, my friends in the Security Service had a problem they had been working on for years. I told them I could do it in two weeks. They were pretty skeptical. So I sat down, grabbed the manuals on what a computer was, learned the assembly language, designed a system, and got it to work in a week and a half. My buddies contacted their commander and told him I was in the wrong career field, that I should be in data automation. That's how I got started 13 years ago." Except for a few practical courses on computer programming techniques or assembly language, Elsen has been self-taught. He joined the Air Force when he was 17, spent 9 years on active duty, and in the last 5 years has been a civilian computer specialist with the 2199th squadron.

Louis C. Elsen

SPACE SATELLITE TRACKS ONE MAN ROWBOAT TRIP FROM BAJA CALIFORNIA TO AUSTRALIA

Dick McCormack
NASA
Washington, DC 20546
202-755-4321
John Kley
Goddard Space Flight Center
Greenbelt, MD 20770
301-344-8141

A British explorer has left the West Coast of North America bound for Australia rowing a rowboat that carries a locater beacon of the National Aeronautics and Space Administration. He is Peter Bird, a 33-year old London photographer. He departed from Baja California on Nov. 11, his destination Brisbane, and his expected voyage 9 to 12 months long.

He is rowing the Britannia II, a 35 foot boat designed for Royal Navy rescue missions. It is self-bailing and self-righting. The beacon is used on various data collection platforms on lakes, icebergs, and other remote locations to measure such factors as water level, ocean currents, etc., in conjunction with the space satellite Nimbus 6 launched in June 1975. The beacon data is transmitted to Greenbelt, MD, computed, and returned to users to pinpoint locations.

Bird expects to communicate via radio with his San Francisco shore station on the 1st and 15th of each month. He will use the Nimbus location data to correct his own position computations. The Nimbus transmitter on Britannia puts out a burst each minute, which identifies its data platform. But Nimbus, in its polar orbit, only "sees" the boat on one orbit per day. Bird's data goes up to Nimbus, down to a ground station in Fairbanks, Alaska, then to Goddard in Maryland, where it is processed, and then it is mailed to Bird's shore station. In times of stress or danger, the station can telephone Goddard for Bird's position.

STRATEGIES AND POLICIES IN INFORMATICS (SPIN): 2ND INTERGOVERNMENTAL CONFERENCE TO BE HELD IN HAVANA, CUBA, SPRING OF 1983

Intergovernmental Bureau for Informatics (IBI)
P.O. Box 10253
00144 Roma, Italy

The government of Cuba has invited the second Intergovernmental Conference on Strategies and Policies in Informatics (SPIN) to Havana, Cuba, in the spring of 1983; and the 10th General Assembly of IBI has accepted.

All member countries of the United Nations will be invited to send official delegations to the Conference.

SPIN II follows five years after the 1st SPIN conference in Torremolinos, Spain. Among the issues which will be considered are:

- economic implications of informatics for industrialization
- educational needs of developing countries
- transborder data flow □

Dwyer – *Continued from page 14*

mer can therefore aim to satisfy some of his user's social needs, or he can fail to. Ignoring them won't make them go away.

Remember that a helpful person is someone who volunteers help. Not someone you have to drag it from. Likewise with programs.

"If the computer were a person, how soon would you hit him on the nose?"

The effect of a well designed dialogue is that the user finds the system easy to learn to use. He does not complain about PITA factors. He is willing and happy to work with it for long periods. One more quote: "Frankly, I get on with the computer better than most of the other people in the office."

Finally, although there is a lot more specific advice I could give on this subject, it is all quite unnecessary. Just understand the basic principle involved. If that computer was a person, how soon would you hit him on the nose? □

Berkeley – *Continued from page 21*

10.	Psychological Trainer	Done. New name, computer-assisted instruction
11.	Automatic Production	Done
12.	Automatic Models of economies, societies, conflicts, etc.	Done

It is true that enormous numbers of applications of these electronic "brains" have been and will be made. A list we published in "The Computer Directory and Buyers' Guide" in 1974 enumerated over 2600 applications. In the future the number of applications of computers will be like the number of applications of books.

Forecasts in 1980

Recently some of my associates and I have found new ways for automatic translation by computer with due regard for meaning. This system applies to:

- automatic computer programming using plain ordinary natural language
- automatic documentation of computer programs
- automatic conversion of computer programs from one language to another
- automatic summarizing of text to varying degrees, etc.

We call this system DJINNI. It applies in a limited context of about 10 to 1000 words. For example 7 lines of English "instructions to a clerk" will change into 67 lines of COBOL program, right the first time. (Reprint of report available on request.)

Many other workers in the field of "artificial intelligence" are producing a body of interesting, remarkable and seminal results.

So I have some 1980 forecasts:

1. More than 50 percent of human programming will vanish as computers take over.
2. Every defined intellectual operation will be performed by computer, faster, better, and more reliably than by a human being.
3. All the language of thought will become calculable like mathematics. □

Shea – *Continued from page 10*

the public is suffering by virtue of AT&T's confinement to regulated industries. H.R. 6121 is therefore unnecessary; it is also unwise. If AT&T is permitted to enter the computer services industry, it should be required to compete. It should not be allowed to shield itself behind its monopoly resources and its control over the communications network. □

Games and Puzzles for Nimble Minds – and Computers

Neil Macdonald
Assistant Editor

It is fun to use one's mind, and it is fun to use the artificial mind of a computer. We publish here a variety of puzzles and problems, related in one way or another to computer game playing and computer puzzle solving, or to programming a computer to understand and use free and unconstrained natural language.

We hope these puzzles will entertain and challenge the readers of *Computers and People.*

NAYMANDIJ

In this kind of puzzle an array of random or pseudorandom digits ("produced by Nature") has been subjected to a "definite systematic operation" ("chosen by Nature"). The problem ("which Man is faced with") is to figure out what was Nature's operation.

A "definite systematic operation" meets the following requirements: the operation must be performed on all the digits of a definite class which can be designated; the result must display some kind of evident, systematic, rational order and completely remove some kind of randomness; the operation must be expressible in not more than four English words. (But Man can use more words to express the solution and still win.)

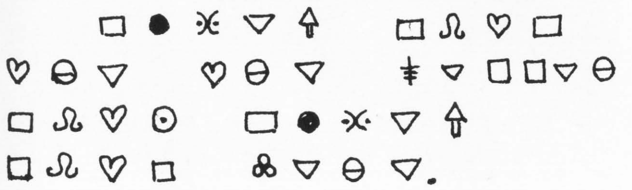
NAMYMANDIJ 8101

```
5 5 9 0 5 8 0 7 1 1 8 2 2 3 7 7 7 2 1 3
3 2 8 4 5 8 9 8 4 1 3 1 4 1 7 8 6 3 9 2
7 6 6 9 5 6 8 9 6 3 8 2 3 4 2 7 4 9 4 5
8 9 4 2 9 9 8 7 7 4 8 8 3 8 7 8 6 3 2 5
0 3 9 8 6 0 7 9 3 5 9 4 8 0 9 0 1 4 5 6
4 2 5 3 4 3 0 8 3 5 4 0 9 3 5 6 7 7 2 8
8 6 1 9 8 5 9 8 6 3 7 6 6 0 4 0 7 0 0 3
5 2 7 7 0 2 0 9 9 2 9 1 5 2 1 6 3 1 8 7
4 4 1 0 0 1 8 7 9 7 2 4 5 9 4 9 1 3 9 5
4 8 0 1 8 9 1 7 5 1 0 4 4 6 2 8 7 4 7 7
```

MAXIMDIJ

In this kind of puzzle, a maxim (common saying, proverb, some good advice, etc.) using 14 or fewer different letters is enciphered (using a simple substitution cipher) into the 10 decimal digits or equivalent signs, plus a few more signs. To compress any extra letters into the set of signs, the encipherer may use puns, minor misspellings, equivalents (like CS or KS for X), etc. But the spaces between words are kept.

MAXIMDIJ 8101



NUMBLES

A "numble" is an arithmetical problem in which: digits have been replaced by capital letters; and there are two messages, one which can be read right away, and a second one in the digit cipher. The problem is to solve for the digits. Each capital letter in the arithmetical problem stands for just one digit 0 to 9. A digit may be represented by more than one letter. The second message, expressed in numerical digits, is to be translated (using the same key) into letters so that it may be read; but the spelling may use puns, or deliberate (but evident) misspellings, or may be otherwise irregular, to discourage cryptanalytic methods of deciphering.

NUMBLE 8101

```
            C L O S E D
         *      F I S T
         --------------
            E S G Y F Y O
          D T C Y O D L
        F F E S G F C
      L G D D S I E            G=O
      ---------------------
  =   L C D L C Y S F F O
```

00453 88603 42414

We invite our readers to send us solutions. Usually the (or "a") solution is published in the next issue.

SOLUTIONS

NAYMANDIJ 8011: Col. 17: one seventh.
MAXIMDIJ 8011: To be young, never talk of the past.
NUMBLE 8011: One dog can't fight.

Our thanks to the following people for sending us solutions: Daniel L. Hirst, Guelph, Ont., Canada – Numble 8009; Helene Patry, Ottawa, Ont., Canada – Numble 8009; Dale Rosenberg, New York, NY – Maximdij 8011; Steve Werdenschlag, Livingston, NJ – Maximdij 8009, Numble 8009, Maximdij 8011, Numble 8011; Chick Daily, Austin, TX – Numble 8009.

The Frustrating World of Computers

by Harry Nelson
1135 Jonesport Ct.
San Jose, CA 95131

SIR, I CAN GUARANTEE IT WONT MAKE THAT MISTAKE AGAIN ——

MAYBE YOU CAN EXPLAIN WHY IT GAVE ME THE BATTING AVERAGES OF THE MAJOR LEAGUES WHEN I ASKED IT FOR OUR SPARE PARTS INVENTORY ——

NO PROBLEM SIR, THE COMPUTER JUST MISPLACED ANOTHER MILLION ——

SURE IT CAN THINK —— IT THINKS I'M STUPID AND I THINK IT'S STUPID ——