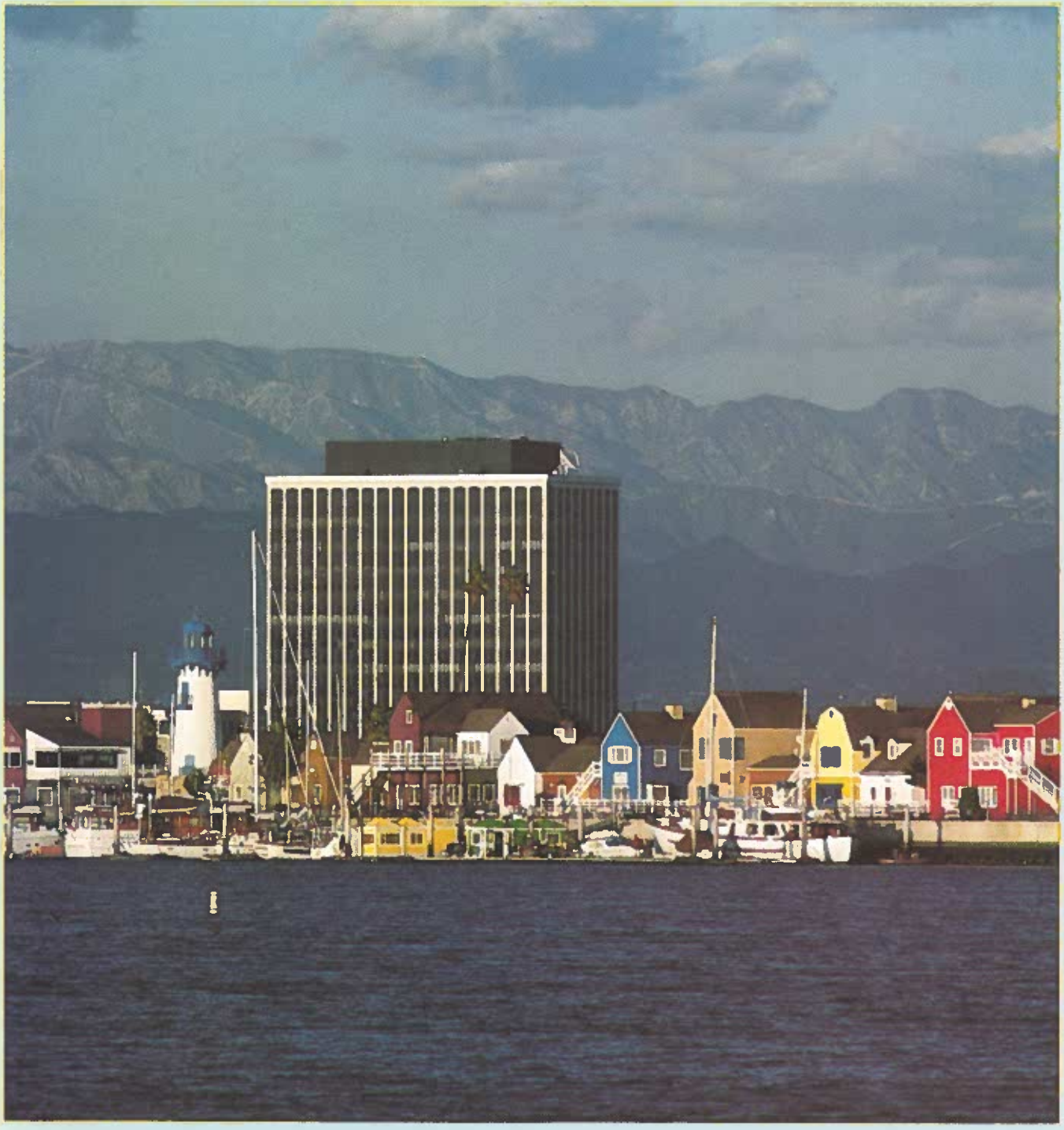
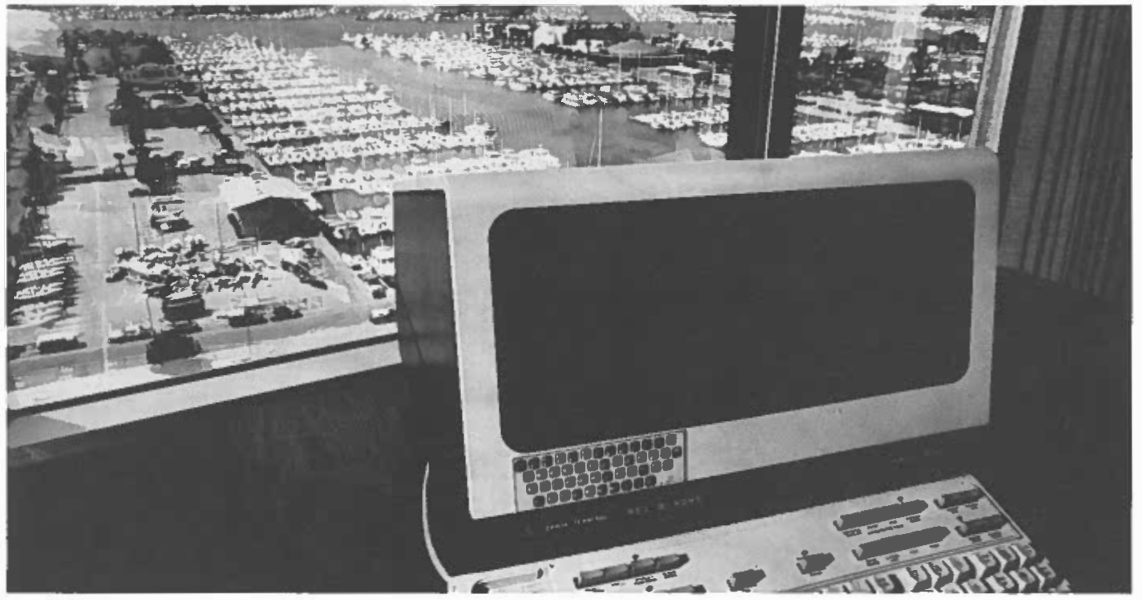
USC/INFORMATION SCIENCES INSTITUTE



Society's need for computer technology is growing at an unprecedented rate. Basic and applied research at ISI is advancing to meet this need, with new projects and changes to the overall research profile reflecting areas of increasing importance. ISI's outstanding staff and facilities, combined with its history of flexibility and continued growth, make it an excellent environment for achieving new successes. We welcome this opportunity to provide a brief view of ISI.

Keith Uncapher

INFORMATION SCIENCES INSTITUTE



Information Sciences Institute is in its tenth year of operation as a center for computer science research. ISI's major goal is to generate and develop new ideas in computer science and to distribute those ideas to the community at large. The Institute combines the activities of its individual projects within a single research environment for its staff members; research is currently concentrated in the areas of artificial intelligence, software system specification, communications and networks, computing environments, and VLSI. Projects share computing facilities including new generation personal computers, large time-shared machines, and ARPANET service through one of the country's largest network nodes.

As one of three major research institutes administered by the University of Southern California, ISI has programmatic autonomy within the university structure. This gives ISI the freedom to identify and engage in significant research programs of its own choice. The Institute maintains a working relationship with USC's School of Engineering, particularly the Departments of Computer Science and Electrical Engineering. ISI benefits from interaction with faculty of USC and other local universities and provides opportunities for research and Ph.D. thesis supervision to graduate students from these universities.

The ISI Environment

ISI's 130 staff members (70 professionals) are organized into the dozen or so individual projects that make up the Institute's interdisciplinary research environment. Project members have significant responsibility for the conduct of their projects and the direction of their research. ISI encourages researcher visibility in the information processing community at large, including publication of research and participation in professional conferences. Performance of professional staff members is gauged in part by their creative influence on the research community outside ISI and their success in identifying new interesting areas of research.

The Institute provides a facilities-rich environment. Each staff member has a terminal for access to shared computing resources; in addition, individual projects provide dedicated personal machines and special hardware dictated by their research needs. Resources also include the Institute's library, hardware development laboratory, and an unusually high level of project and administrative support. University of Southern California facilities are available to all staff members.

ISI overlooks the Pacific Ocean from Marina del Rey, a suburban recreational community containing the world's largest man-made small boat harbor. Twenty minutes from downtown Los Angeles and USC, the Marina is far from the smog in the midst of some of the most desirable living areas in the Los Angeles basin. The full diversity of Los Angeles' ethnic, cultural, and recreational activities are within easy reach.

Tenth Anniversary

In May 1982, ISI celebrates its tenth anniversary. We are commemorating the event with a two-day seminar to look to the next ten years in the information sciences. Each of the following speakers will present his personal perspective of opportunities, goals, and responsibilities.

Michael Dertouzos, Director, Laboratories for Computer Science, MIT;
Edward Feigenbaum, Professor, Stanford University;
Robert Kahn, Director, Information Processing Techniques Office, DARPA;
Robert Lucky, Director, Electronic and Computer Systems Research, Bell Laboratories;
Steven Lukasik, Chief Scientist, Federal Communications Commission;
Allen Newell, University Professor, Carnegie-Mellon University;
Lawrence Roberts, President, Subscriber Network Products, GTE TELENET Communications;
Ivan Sutherland, President, Sutherland, Sproull & Associates, Inc.

Artificial Intelligence

The field of artificial intelligence is beginning to have major impact on how computers are used to solve problems. ISI projects are investigating fundamental problems in the representation and use of knowledge by computers as well as the experimental application of AI technology to real-world problems. Current activity is focused in the following areas:

Inference
- Control of reasoning processes that must deal with large amounts of knowledge
- Tradeoffs between detailed knowledge representation and sophisticated control structure
- Plausible inference and the representation of uncertain knowledge

Knowledge base design
- Consistent representation of various kinds of knowledge
- Maintenance of large knowledge bases
- Acquisition of domain-specific knowledge

Expert systems
- Architectures for expert systems
- Modeling of expert knowledge for delivery to nonexperts

Natural language
- Generation of English text
- Adaptation of parsers to new domains of discourse
- Informal specifications

Several projects contribute their individual viewpoints to this ongoing research. The *Consul* project is examining the use of knowledge-based inference to provide a domain-independent cooperative interface to a set of interactive services. The *Control of Expert Systems* project is using the Hearsay-III domain-independent architecture to develop better ways of providing the flexible control needed in expert systems. *SAFE* investigates the use of built-in process-description knowledge and dynamically acquired domain knowledge to produce formal program specifications from informal descriptions. The *Knowledge Delivery* project is applying the results of basic linguistic research to the problem of constructing multiparagraph English text.

Communication and Networks

Communication among computers has become an important part of computer science research, not only as a topic itself, but also in support of other areas. ISI is working to improve computer communications through the development of networks and network protocols ranging from office environments to international networks.

Applications

Advanced applications systems use integrated multimedia data for interactive conferencing, briefing aids, and computer mail. The user should be given choices among media including text, voice, graphics, facsimile, and video.

Environments

The communication environment is an interconnected system of networks of many different types, including long-haul (ARPANET), satellite (SATNET, WBNET), radio (PRNETs), and local (LCSNET, ETHERNET). A user and his computation processes should be able to access facilities across this entire environment.

The *Internet Concepts* project is working in the areas of protocol design, application, and verification. The *Wideband Communication* project is working in the areas of speech and video digital communication using a broadcast packet satellite network. The *Command and Control Graphics* project is working on device-independent color graphics to support command and control systems. The *Information Processing Center* supports a variety of computer communication facilities and programs; the *New Computing Facility* project will provide the framework for the next generation of these.

Computing Environments

ISI's interest in computing environments spans the full range of interactive facilities, from the maintenance of a large service-oriented ARPANET node to experiments with a coherent environment for providing natural language interaction with the user. The collective goal of ISI projects pursuing research in this area is a deeper understanding of three aspects of cooperative interaction in computing environments:

Cooperation among services

Unification of diverse hardware (including a variety of personal machines) and system software facilities into a single computing environment that is coherent from a user software viewpoint.

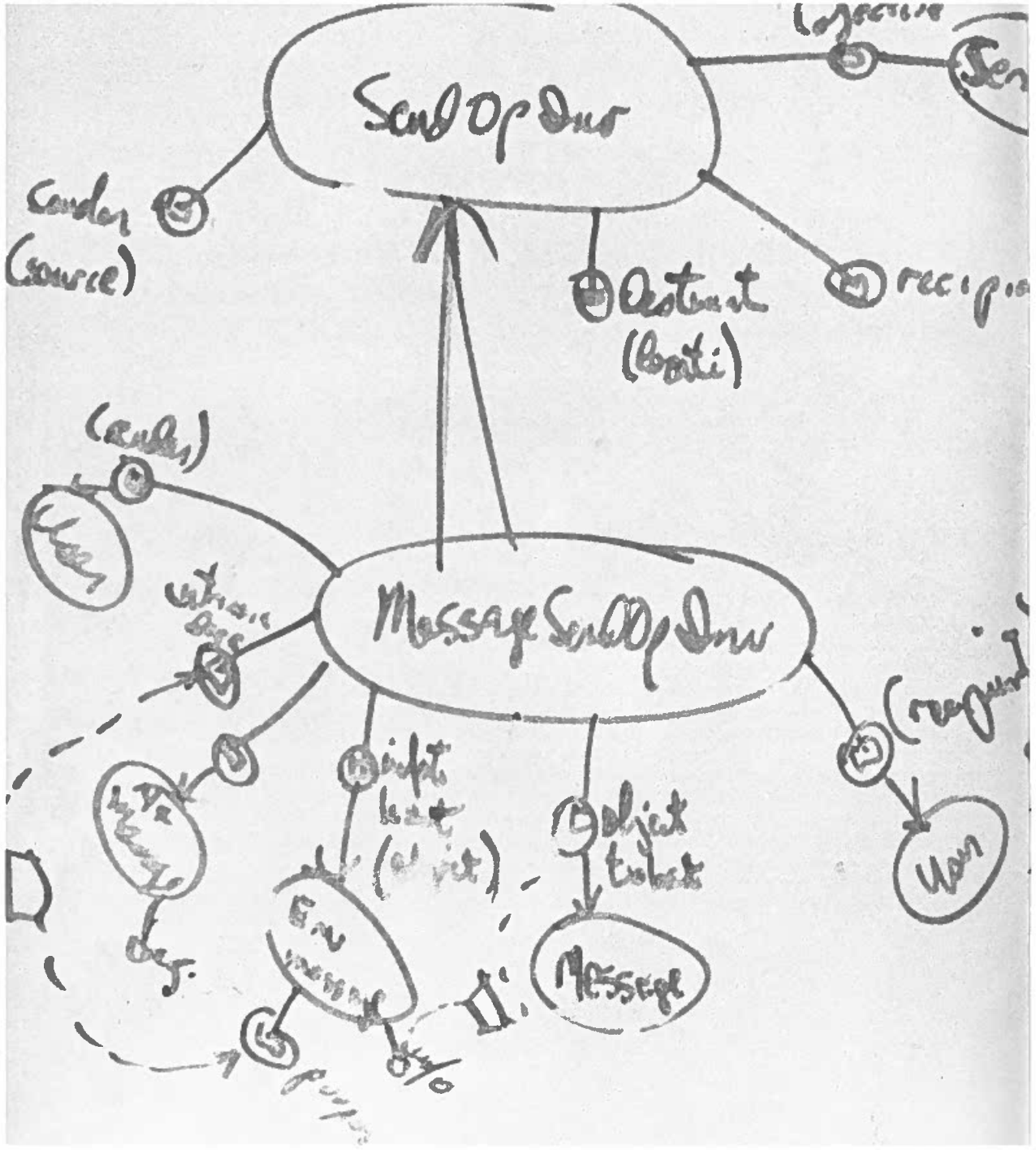
Cooperation among users and services

Creation of a single interface to all services in the environment that interacts naturally with the users (understanding requests and providing help).

Cooperation among users

Support for group efforts in program development, document preparation, and other office and administrative interactions, i.e., a systemwide information management facility.

The *Information Processing Center* provides stable, continuous support facilities. The *New Computing Facility* project is planning for near- and far-term advances to support the needs of ISI's research community. *Consul* is attempting to provide a uniform, friendly user interface, using knowledge-based inference; the emerging *Information Management* and *CUE* projects are investigating other aspects of the uniform interface problem. *Command and Control Graphics* concentrates particularly on graphical interfaces. The *Wideband Communication* and *Internetwork Concepts* efforts deal with how to link computers effectively. The *Interlisp* project is developing an easily portable version of the language and its support system.



Cooperative Interactive Systems (Consul)

William Mark, David Wilczynski, Thomas Lipkis, and Bill Swartout

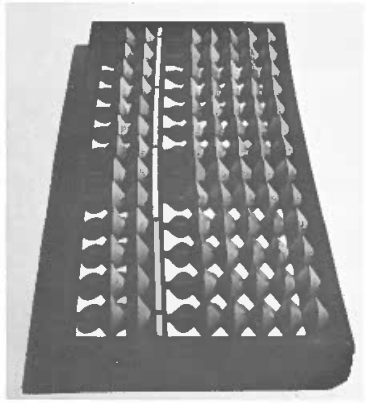
Current interactive systems must be made more habitable for a wide variety of users, especially those having little experience with computing. We are currently building the **Consul** system to explore methods for allowing natural interaction--natural-language requests, explanation of system facilities, user-understandable error handling--between a user and a set of online services (for text manipulation, message handling, etc.). Consul works by mapping user requests into appropriate system actions as dictated by a detailed model of system capabilities and requirements. This model is based on built-in service-independent knowledge, but is tailored to individual services via a dialogue-driven knowledge acquisition process. As each piece of a new service is built and added to Consul, the system engages in a dialogue with the service builder to determine how the piece fits into its model of the service so far and of interactive systems in general. Our research therefore includes not only modeling and mapping methods that generalize across services, but also a programming environment for service builders that integrates their services into Consul's knowledge base.

Specification Formulation (SAFE), Testing (Gist Validation), and Implementation (TI)

Bob Balzer, Don Cohen, Lee Erman, Martin Feather, Neil Goldman, Jack Mostow, Bill Swartout, Dave Wile, and Steve Fickas

This project comprises three interrelated efforts working to facilitate the creation, testing, and optimized implementation of program specifications. The first effort, called *Specification Acquisition From Experts (SAFE)*, is directed at helping people create unambiguous, consistent, and complete formal program specifications through informal description. While end users are quite capable of providing informal process-oriented descriptions of the task being automated, formalisms of any kind provide major impediments. The informal descriptions are characterized by partial, rather than complete, constructs. The system uses a knowledge base of program well-formedness rules to disambiguate and complete the informal natural language descriptions. An early version of the system has successfully converted several small informal specifications into formal specifications. Attention is now focused on handling large specifications through incremental formalization; this new implementation uses *Hearsay-III* as its base.

The second effort seeks to ensure that a formal specification matches the user's intent, by systematically investigating its behavior through symbolic execution. The tasks here include the design of a formal abstract specifica-



tion language (called *Gist*) and mechanisms for incremental specification based on elaboration, and the development of techniques for symbolic execution (some of which have already been developed within the SAFE context) and for explanation of the simulated behavior.

The third effort, called *Transformational Implementation*, seeks to provide a methodology guaranteed to produce only valid implementations of the formal specification by involving the computer in the implementation process to ensure that the evolving implementation remains consistent with the original formal specification. This is accomplished by limiting the activity of programmers to a choice of optimizations. These optimizations are embodied in a catalog of transformations, applied by the system in response to the programmer's choices, after verifying that they are applicable. A prototype grammar-based system has been implemented; current efforts are focused on structuring the steps of the development process and on facilitating the highest level optimization decisions--those that implement the freedoms provided by the specification language. Such a system supports multilevel design by mechanizing the movement (under user guidance) between levels. We are exploring the generality of this approach by attempting to apply it to VLSI design, which employs different design considerations and implementation techniques.

Control of Expert Systems

Lee Erman and Jeff Barnett

Expert, knowledge-based systems for particular problem domains (e.g., medical diagnosis, chemical analysis, and traffic control) are becoming increasingly important, both as practical tools and as a major area of AI research. This project is concerned with the architecture of such systems and, in particular, with their *control*--the division and allocation of resources during the execution of the expert system. An expert system often requires *soft control*--a flexibility of interaction among the various pieces of domain knowledge that allows for opportunistically allocating resources to activities most likely to make efficient progress. The solution to this problem itself requires the application of knowledge, both general-purpose (e.g., "A faster technique is preferred to a slower one") and domain-specific (e.g., for a medical consultation system: "An external test is preferred to an invasive one"). The conceptual and operational framework for this project is *Hearsay-III*, a system we have developed for building and experimenting with expert systems.

Knowledge Delivery in Multiparagraph Text

William Mann, Yasutomo Fukumochi, Steve Klein, and Christian Matthiessen

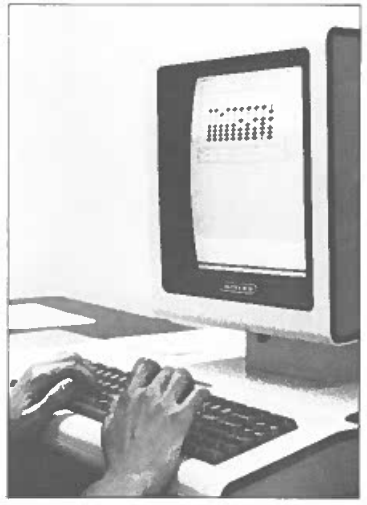
This project is developing new methods for autonomous text composition by machine, with the focus on generation of fluent English. The text generator, called *Penman*, is being developed to explore generation in multiple knowledge domains, including description of programs and program operations. Penman will seek to deliver knowledge (in English) from inside a system that was not originally designed to have a knowledge delivery component.

Penman contains a large systemic generative grammar of English. This grammar system extends the framework of M. A. K. Halliday with new semantic components and methods for interacting with a knowledge representation external to the grammar.

Formal Semantics

Vittal Kini, Dave Martin, and Allen Stoughton

The Formal Semantics project is carrying out research into methods for the validation and use of formal semantic definitions of programming languages. The Ada Formal Semantic Definition (FSD), written by the French group at INRIA, is being used as a vehicle for this research. A program which interprets the Ada FSD will be used as a tool to validate the FSD. The interpreter with the FSD installed within it will comprise a semantic-checker which takes Ada programs as input and either indicates where and why a program does not conform to the language semantics or else provides as output the denotations of the program. In the course of constructing the semantic-checker, the FSD will need to be parsed and type-checked, thus ensuring that syntactic and other minor errors in the FSD are revealed. It is expected that the semantic-checker will be employed in two phases. In the first phase, Ada programs, with known and well-understood intent, will be used as test case inputs. This phase will exercise the FSD, help to uncover deeper and less obvious errors in it, and instill confidence in its correctness. In the second phase, the checker may be used in answering detailed questions of interpretation of the language semantics. In addition to benefitting the definition of Ada directly, the project will provide insights into improving the state of the art in defining the formal semantics of practical programming languages.



New Computing Facility

Dan Lynch, Danny Cohen, Joel Goldberger, Jim Koda, Jon Postel, and Craig Rogers

This effort will provide significantly improved computing facilities for researchers at ISI, as well as for remote and mobile users. Previous facilities have been almost exclusively large time-shared mainframes with individual file systems. The major goal of this project is to give each researcher a powerful, dedicated personal machine upon which he or she can perform most research and administrative computing. Only when the size of the task significantly exceeds the capabilities of the personal machine will the researcher resort to a large central server. There will be a single common file system, high-speed communication among all the computers, and common servers for other specialized tasks such as printing and external communication. Another goal is to allow new machines to be integrated into the facility with minimal effort. A key issue currently under consideration is the selection of a user interface suitable for the needs of ISI's research community.

Internetwork Concepts

Jon Postel, Danny Cohen, Greg Finn, Alan Katz, Paul Mockapetris, David Smallberg, and Carl Sunshine

Many packet-switched computer communication networks now exist; this project is investigating ways to interconnect them usefully. The project has three task areas: the formal analysis of protocols, the design and prototype implementation of internetwork applications, and the design of protocols and development of new communication concepts.

The protocol analysis area is focusing on the correctness (in the program verification sense) of the Transmission Control Protocol (TCP). TCP is the host-to-host reliable data stream protocol developed for the internet environment. Several program and protocol analysis tools are being explored.

The protocol applications area is focusing on computer mail. Two distinct systems are being developed: a text multinetwork mail system for use in the short term, and a multimedia mail system for use in the long term. Other applications will be explored in later stages of the project.

The protocol design and concepts area is focusing on the host-to-host and gateway level protocols, particularly the Internet Protocol (IP), a datagram protocol which is universal in the internet environment. There are many design issues to be resolved--for example, very mobile hosts, controlled routing, access control, fault isolation, and resource allocation.

Wideband Communication

Steve Casner, Bill Brackenridge, Danny Cohen, Randy Cole, and Ian Merritt

After several years of research with relatively low bandwidth packet networks, ISI is participating in a *Wideband Communication Program* (initiated by ARPA and the Defense Communications Agency) to investigate packet switching on a high-bandwidth satellite network. One goal of this project is to develop the technology required to support future packet speech systems with thousands of voice channels, beginning with an experimental facility that can be accessed by a much broader user community than that of the first demonstration systems; this will help to determine the feasibility of realizing the economies of integrated voice/data packet transmission. A second goal is to explore new modes of packet communication made possible by the increase in bandwidth of the satellite network over that of previous packet networks and to investigate how the added bandwidth can benefit old modes of communication. We are building systems for transmission of narrowband packet video, multiple channels of packet speech, graphics, text, and bulk data.

Command and Control Graphics

Richard Bisbey, Ben Britt, Dennis Hollingworth, and Pamela Finkel

The Command and Control Graphics project is developing a distributable, display-device-independent vector graphics system for use in a command and control environment. The user communicates with the system via a device-independent graphics language whose primitives are mapped to internal graphics protocols from which device-specific graphics orders are generated. The project is also developing graphics command and control application programs, including a Situation Display (naval force information displayed on a geographic background) and a Briefing Aid system that demonstrate the use of the graphics system. The software is currently in use at the Naval Ocean Systems Center as part of the Navy's Advanced Command and Control Architectural Testbed, as well as at sites connected by the ARPANET and the secure subnet.

VLSI Implementation System

Danny Cohen, Yehuda Afek, Ron Ayres, David Booth, Victor Brown, Joel Goldberg, Lee Richardson, Barden Smith, and Vance Tyree

The VLSI project is focused on research in the design methodology of VLSI circuits at various levels and on the development of the tools needed for providing fast turnaround fabrication service to the ARPA VLSI research community.

Research is progressing in the automatic generation of chips from purely functional (nongeometric/algorithmic) specification. Two distinct approaches and their integration are being pursued. First, we are working on a silicon compiler, called *Bristle Blocks*, which produces a limited class of chips, namely microprocessors, with remarkably efficient designs, competitive with manual designs. Second, we are pursuing a silicon compiler, *RELAY*, which accepts the specification of arbitrary synchronous systems in terms of hierarchically described synchronous logic equations. We note that the choice of appropriate specification language is of fundamental importance and that the target medium (silicon) has a character quite different from that of software.

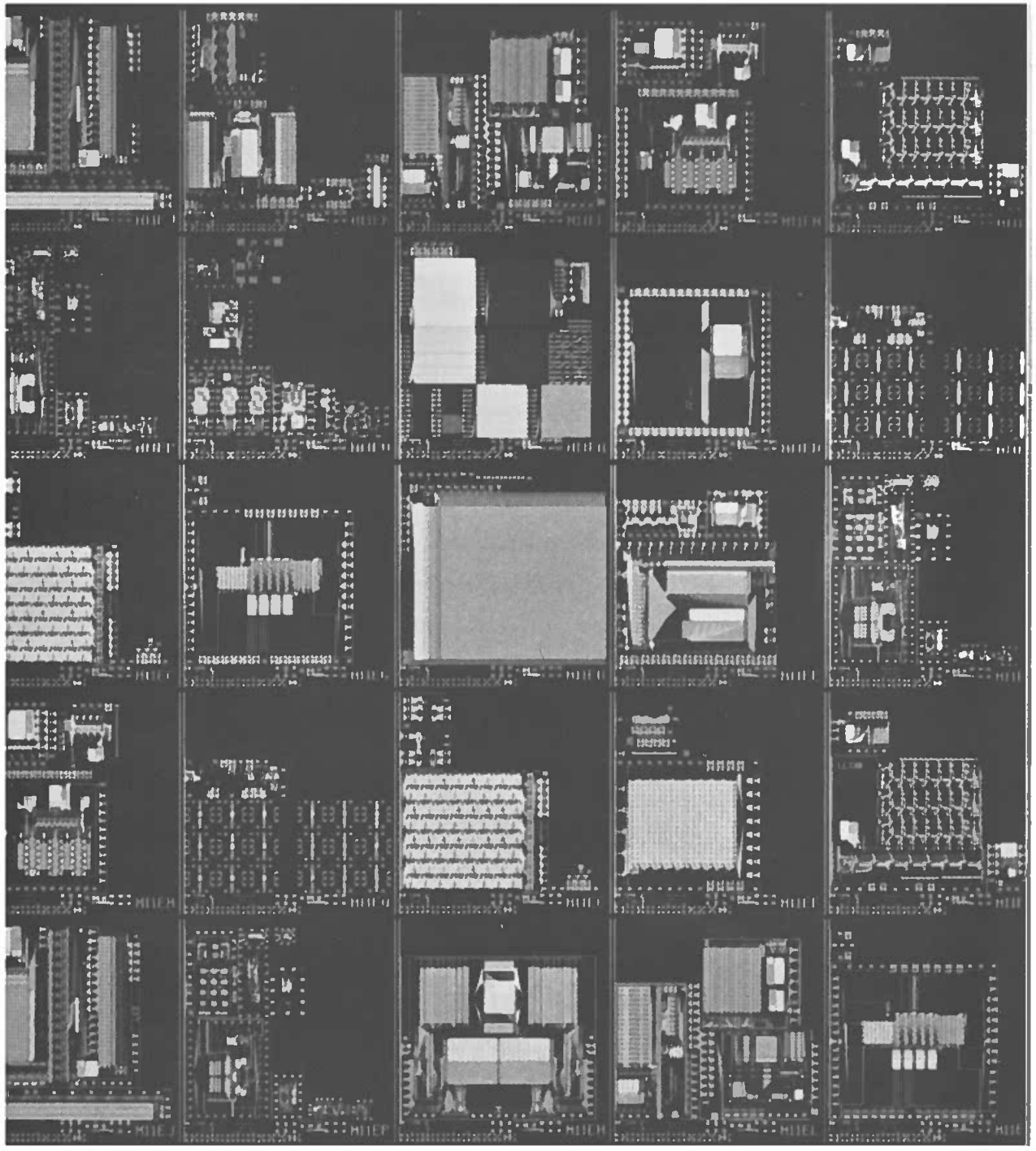
We are also developing the concepts involved in managing useful libraries for popular subdesigns ("procedures"). Unlike software libraries, useful silicon libraries require many implementations of the same function. The geometric alternatives presented in multiple implementations provide for the possibility that people will find a design that fits nicely within the geometric constraints imposed by other parts of their design.

ISI is also developing and maintaining *MOSIS*, which supports the fast turnaround fabrication requirements of the ARPA VLSI research community. MOSIS handles many of the issues of information processing, geometric manipulations, quality control (by wafer probing and device testing), and general management. In addition to the management-oriented tasks, the MOSIS team is working on the development of standard acceptance tests and other facilities required for developing the interfaces needed to support the separation of the design from the fabrication process.

Interlisp-VAX Implementation

Dan Lynch, Ray Bates, Dave Dyer, Andrea Ignatowski, Steve Saunders, and Don Voreck

This project is implementing and will maintain a fully compatible, portable, large address-space Interlisp. The first version is implemented in C under UNIX and VMS for the VAX computer and was released in March 1982.



In addition to its ongoing research efforts, ISI is in the process of exploring new research directions that will likely develop into full-fledged projects. Exploratory work is currently under way in the areas of integrated interactive system design and information management.

Consistent Underlying Environment (CUE)

Tom Kaczmarek et al.

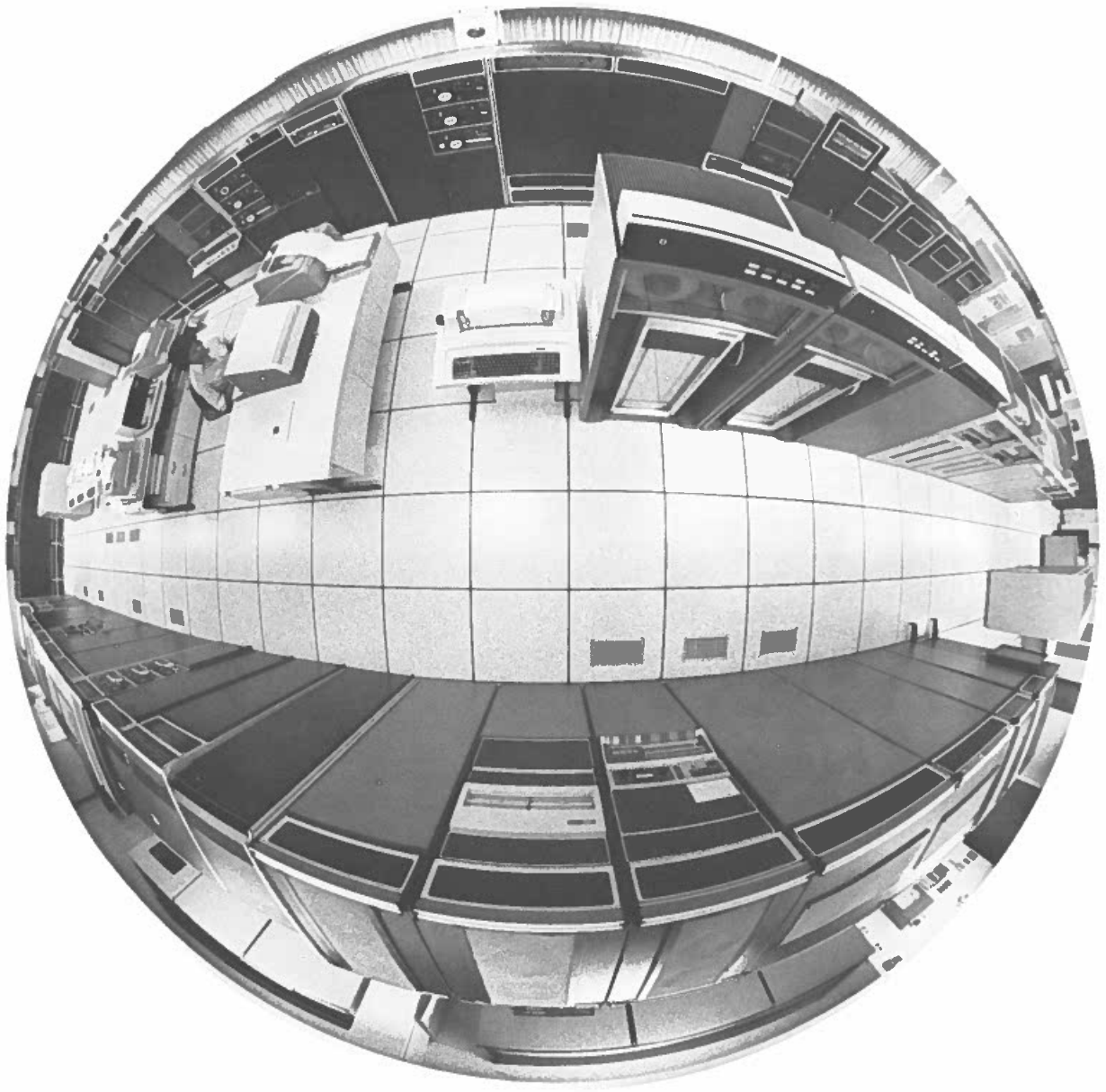
The Consistent Underlying Environment (CUE) project will examine the problem of constructing interactive systems whose various services (e.g., electronic mail, online appointment calendar, word processors) can be used in an integrated way to perform user tasks. For example, the task of sending a message to all the attendees of a particular meeting is quite difficult in current interactive systems, in which the mail and calendar services are totally separate subsystems. The goal of the CUE project is to create a methodology in which all of the services in the machine can be built into and accessed within a single interactive environment.

Information Management

Bob Balzer, Dave Dyer, Michael Fehling, and Steve Saunders

The aim of this new project is to build a support environment that facilitates the creation, integration, and evolution of computational services required by a community of interacting users. This environment will be based on uniform mechanisms for defining and instantiating information structures, building and modifying relationships among them, searching for those that satisfy predicates, establishing and maintaining consistency among objects, building services through event-based agents, and examining the behavior of such services. A key objective of this project is to develop techniques which allow such an environment to operate at sufficient speed to be practical.

The uniformity of the mechanisms on which the environment is based, and the simplicity of the underlying computation model (constrained objects), should enable users to comprehend and modify services created by others. This hypothesis will be tested by building the environment and a kernel set of services, and then extending them.

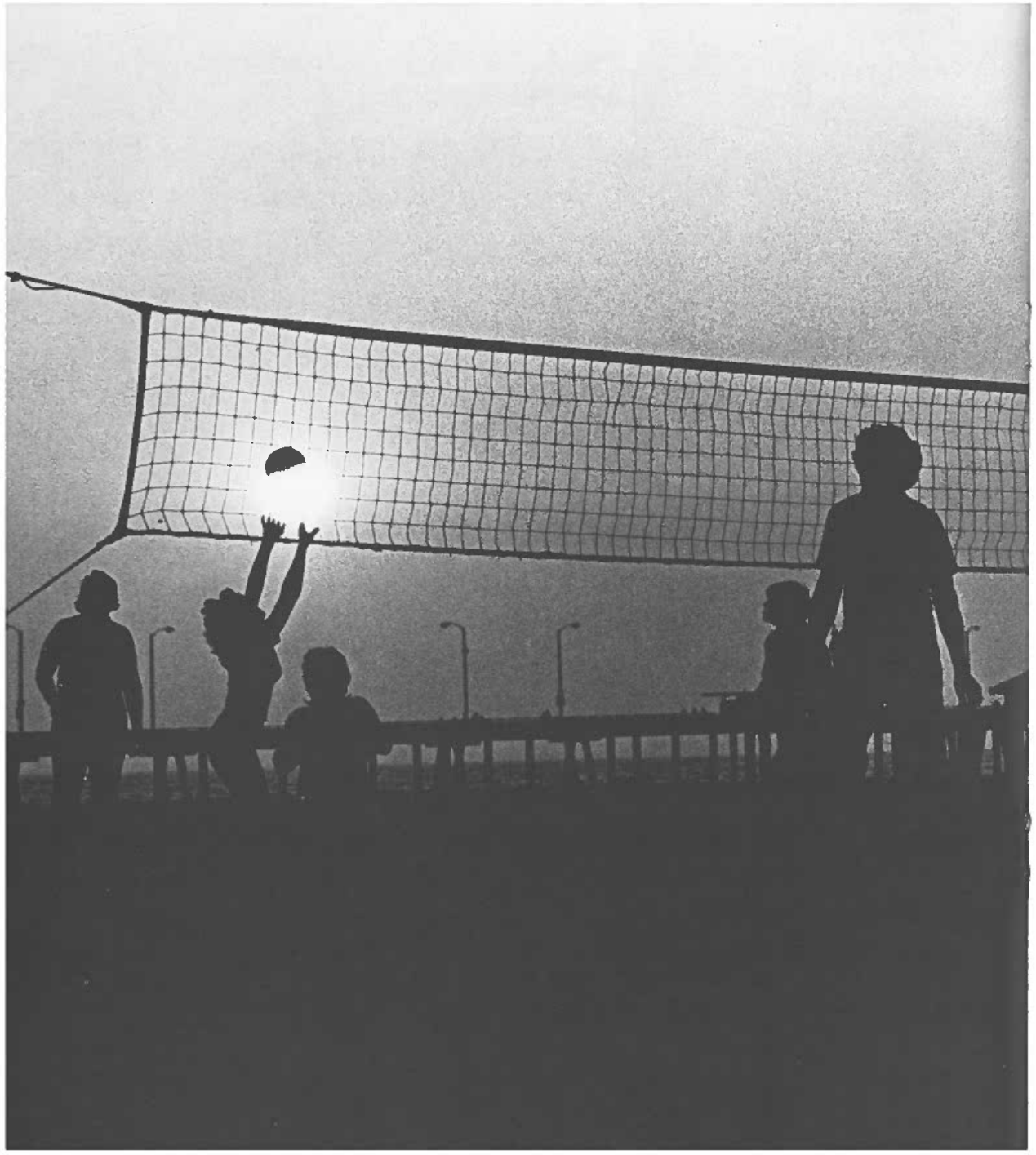


FACILITIES

Available and reliable computing power is a basic necessity for performing research in this field. ISI's **research computing facilities** currently comprise advanced personal computers (Xerox Dolphins, Symbolics LISP Machines, Three Rivers Perqs), Digital KL-10s, VAX 11/780s, PDP-11 minicomputers, and specialized devices for research in areas such as graphics, speech, and emulation. ISI's computing environment is currently undergoing major revision in response to changing research needs. The goal is to provide each user with a substantial increase in address space and computer cycles (both personal and time-shared) and advanced terminals for office and home use.

The **hardware development lab** is responsible for constructing any special equipment required for ISI research. The lab contains machine shop and electronics tools, including an online prom-burning facility and a prototype printed circuit board fabrication capability (currently under development). Past projects have included video recording systems, ARPANET and Ethernet interfaces, and equipment for digital transmission of speech and video. State-of-the-art prototype products have also been delivered to several universities and government facilities; lab developments have directly influenced new products at Hewlett-Packard and Digital.

Staffed by a full-time technical librarian, ISI's **library** features an extensive collection of books and current technical reports, subscriptions to more than one hundred technical journals, and online access to several bibliographic databases. The up-to-date collection covers the full range of computer science, with special depth in those areas of particular concern to ISI's researchers. The **publications department** has two full-time editors to assist in the preparation of proposals, articles, and conference presentations, and to publish and distribute technical reports and manuals. Additional services range from answering questions about document-preparation systems to completely rewriting the transcript of a conference talk into an appropriate form for publication. A full-time **graphics** designer provides expertise in designing and producing visual materials to supplement text and oral presentations. The graphics department also provides photographic, layout, and drafting support for project research.



STAFF

Keith Uncapher, Executive Director
Thomas O. Ellis, Deputy Director
Robert Blechen, Institute Administrator

Research Staff

Ronald Ayres
B.A., Mathematics, California Institute of Technology, 1974
Ph.D., Computer Science, California Institute of Technology, 1979
Language processing, software and hardware compilers, recognizing fundamental properties of information as they relate formal and informal beings

Robert Balzer
B.S., Electrical Engineering, Carnegie-Mellon University, 1964
M.S., Computer Science, Carnegie-Mellon University, 1965
Ph.D., Computer Science, Carnegie-Mellon University, 1966
Transformation-based programming, formal specification, expert systems, design, informal language

Jeffrey A. Barnett
Methods of plausible reasoning, control structures for AI systems, representations of knowledge, theory and development of programming languages, theory of distributed evaluation, combinatorics

Raymond L. Bates
A.B., Mathematics, University of Southern California, 1973
M.S., Computer Science, University of Southern California, 1980
Specification and verification methods, software production technology, programming languages

Richard L. Bisbey
A.B., Mathematics, University of California, Los Angeles, 1967
M.S., Information Systems, University of California, Los Angeles, 1968
Graphics

William A. Brackenridge
Wideband communication

Ben Britt
B.S., Electrical Engineering, University of Southern California, 1975
M.S., Computer Science, University of Southern California, 1976
Graphics

Stephen L. Casner
B.A., Mathematics, Occidental College, 1973
M.S., Computer Science, University of Southern California, 1976
Real-time multiprocess system design, network protocols, voice and video bandwidth compression, restricted domain programming

Danny Cohen
B.Sc., Mathematics, Technion, Israel Institute of Technology, 1963
Ph.D., Mathematics/Computer-Science, Harvard University, 1969
VLSI, machine architecture, real-time communication, computer communication protocols, computer image generation, cockpit information/flight simulation

Don Cohen
B.S., Mathematics, Carnegie-Mellon University, 1973
Ph.D., Computer Science, Carnegie-Mellon University, 1980
Reasoning, learning, specification languages

E. R. (Randy) Cole
Professional Engineer, Mathematics, Colorado School of Mines, 1967
Ph.D., Electrical Engineering (Computer Science), University of Utah, 1973
Speech processing, image processing, fast signal-processing computers, computer graphics

Alvin S. Cooperband
A.B., Biochemistry, Harvard University, 1956
M.A., Psychology, University of California, Los Angeles, 1968
Human-machine interaction, user interfaces, personal computing

David Dyer
B.A., Mathematics, Immaculate Heart College, 1975
User interfaces, very large databases

Thomas O. Ellis
B.S., Engineering, University of California, Los Angeles, 1951
M.S., Engineering, University of California, Los Angeles, 1953
Human-machine communication, hardware systems architecture

Lee D. Erman
B.S., Mathematics, University of Michigan, 1966
M.S., Computer Science, Stanford University, 1968
Ph.D., Computer Science, Stanford University, 1974
Architectures for and control of knowledge-based systems, human/machine expert systems, informal specification, distributed artificial intelligence, speech-understanding systems

Martin S. Feather
B.A., Mathematics and Computer Science, Cambridge University, 1975
M.A., Mathematics and Computer Science, Cambridge University, 1976
Ph.D., Computer Science and Artificial Intelligence, University of Edinburgh, 1979
Methodologies for program development and maintenance, computer-assisted program implementation, program specification, program transformation

Michael R. Fehling
B.S., Mathematics and Psychology, University of Illinois, 1968
Ph.D., Cognitive Psychology, Stanford University, 1973
Learning and knowledge representation, computational linguistics, software psychology

Gregory G. Finn
B.S., Physics, Brandeis University, 1973
M.S., Computer Science, University of Southern California, 1977
Computer networking, multimedia message systems

Yasutomo Fukumochi
B.A., English Literature, Ritsumeikan University, Kyoto, 1973
M.A., Linguistics, University of California, Los Angeles, 1980
Natural language processing, linguistics, semantics

Louis Gallenson
BSEE, Electronics, University of California, Berkeley, 1951
Local networks, system architecture, terminals, CPU I/O devices, emulations and microprogrammable CPUs

Joel Goldberg
B.S., Chemistry, California Institute of Technology, 1967
M.S., Computer Science, University of Wisconsin, 1969
Emulation and emulation support systems, graphics, distributed computations

Joel Goldberger
User interfaces, personal computing, distributed networks, office automation

Neil M. Goldman
B.S., Mathematics, Stanford University, 1969
M.S., Computer Science, Stanford University, 1971
Ph.D., Computer Science, Stanford University, 1973
Formal specification languages, computer aids for program and specification synthesis, computer generation of natural language

Dennis Hollingworth
B.A., Mathematics, University of California, Riverside, 1967
Graphics

Andrea Ignatowski
B.S., Computer Science, University of Southern California, 1982 (expected)
Artificial intelligence, knowledge representation, software engineering

Thomas S. Kaczmarek
B.S., Mathematics, University of Wisconsin-Whitewater, 1968
MSEE, Computer and Information Sciences, University of Pennsylvania, 1973
Ph.D., Computer and Information Sciences, University of Pennsylvania, 1977
Knowledge representation, user interface, speech, graphics

Vittal Kini
B. Tech., Electrical Engineering (Electronics), Indian Institute of Technology, Bombay, 1973
M.S., Electrical Engineering (Computer Engineering), Carnegie-Mellon University, 1975
Ph.D., Electrical Engineering, Carnegie-Mellon University, 1981
Program verification, programming language semantics, hardware reliability, hardware design automation

Steve Klein
B.S., Engineering, California Institute of Technology, 1976
Natural-language generation and understanding, knowledge representation, VLSI design systems, hardware representation systems

Jimmy T. Koda
B.A., Mathematics, University of Southern California, 1975
Distributed systems, networks, databases

Thomas A. Lipkis
B.S., Computer Science, University of Southern California, 1978
Ph.D., Information and Computer Science, University of California, Irvine (in progress)
Automated explanation of expert systems, knowledge representation and modeling in expert systems

Daniel C. Lynch
B.S., Mathematics, Loyola University, Los Angeles, 1963
M.A., Mathematics, University of California, Los Angeles, 1965
Operating system performance analysis and tuning, computer networking, management of advanced computing environments

William C. Mann
B.S., Electrical Engineering, Lehigh University, 1956
M.E.A., Engineering Administration, George Washington University, 1963
Ph.D., Computer Science, Carnegie-Mellon University, 1973
Natural language processing, artificial intelligence, knowledge representation, linguistics, natural inference

William S. Mark
S.B., Computer Science, Massachusetts Institute of Technology, 1974
S.M., Computer Science, Massachusetts Institute of Technology, 1974
E.E., Electrical Engineering, Massachusetts Institute of Technology, 1975
Ph.D., Computer Science, Massachusetts Institute of Technology, 1976
Representation of inferential knowledge, expert systems

David F. Martin
(Professor of Engineering and Applied Sciences, UCLA)
B.S., Engineering, University of California, Los Angeles, 1960
M.S., Engineering, University of California, Los Angeles, 1962
Ph.D., Engineering, University of California, Los Angeles, 1966
Semantics of computation, semantics of programming languages, correctness proofs of programming systems implementation

Ian H. Merritt
Communications, information systems, telephony and radiotelephony, human behavior

James Moore
S.B., Mathematics, Massachusetts Institute of Technology, 1964
Ph.D., Computer Science, Carnegie-Mellon University, 1971
Natural language, AI/knowledge engineering

David (Jack) Mostow
A.B., Applied Mathematics, Harvard College, 1974
Ph.D., Computer Science, Carnegie-Mellon University, 1981
Heuristic program synthesis, transformational implementation, formalizing AI methods that use domain knowledge, mapping behavioral specifications to programs and VLSI designs

Robert H. Parker
S.B., Electrical Engineering, Massachusetts Institute of Technology, 1969
M.Eng., Electrical Engineering, Cornell University, 1970
Communications, human-machine interaction, hardware design

Jon Postel
B.S., Engineering, University of California, Los Angeles, 1966
M.S., Engineering, University of California, Los Angeles, 1968
Ph.D., Computer Science, University of California, Los Angeles, 1974
Computer communication protocols, interprocess communication, distributed applications

Leroy C. Richardson
B.S., Electrical Engineering, Carnegie-Mellon University, 1965
M.S., Electrical Engineering, Carnegie-Mellon University, 1966
Ph.D., Electrical Engineering, Carnegie-Mellon University, 1972
Engineering, applications programming

Craig Milo Rogers
Computer networking, distributed operating systems, programming environments, real-time programming

Steven E. Saunders
S.B., Electrical Engineering, Massachusetts Institute of Technology, 1972
Ph.D., Computer Science, Carnegie-Mellon University, 1979
Design of pocket-size personal computers and their software, flat-panel displays, digital sound synthesis, customized compact interpretive code, authentication and encryption via public-key systems

Barden E. Smith
B.S., Electrical Engineering, University of Utah, 1979
VLSI, computer-aided design systems, data communications

Allen Stoughton
B.S., Mathematics/Computer-Science, University of California, Los Angeles, 1979
M.S., Computer Science, University of California, Los Angeles, 1981
Protection models of operating systems, semantics of programming languages

Carl Sunshine
B.A., Physics, University of Chicago, 1971
M.S., Computer Science, Stanford University, 1973
Ph.D., Computer Science, Stanford University, 1975
Computer networks, communication protocol design and analysis, network interconnection

William Swartout
B.S., Mathematical Sciences, Stanford University, 1974
S.M., Computer Science, Massachusetts Institute of Technology, 1977
Ph.D., Computer Science, Massachusetts Institute of Technology, 1981
Computer-generated explanations of programs, computer-aided program implementation, knowledge representation, knowledge-based systems

Vance C. Tyree
BSEE, Electrical Engineering, University of California, Berkeley, 1965
MSEE, Electrical Engineering, University of California, Berkeley, 1966
VLSI design automation systems, fault-tolerant VLSI circuitry, highly concurrent processors on silicon

Donald Voreck
B.S., Electrical Engineering, Rutgers University, 1980
Artificial intelligence, applications programming, problem solving

David Wilczynski
B.A., Mathematics, University of California, Los Angeles, 1968
M.S., Computer Science, University of Southern California, 1972
Ph.D., Computer Science, University of Southern California, 1975
Intelligent user interfaces, knowledge-based systems, software design methodologies

David S. Wile
Sc.B., Applied Mathematics, Brown University, 1967
Ph.D., Computer Science, Carnegie-Mellon University, 1974
Programming language design, specification languages, programming methodologies, programming environments

Graduate Research Assistants

Yehuda Afek
David Booth
Steve Fickas
Alan Katz
Christian Matthiessen
Paul Mockapetris
David Smallberg
Craig Taylor

Management and Administration

Administrative Support
Suzanne Robb
Joyce K. Reynolds
Janet O'Hare
Phyllis Keiser
Roylene Jones

Research Project Support
Sharyn Brache
Victor Brown
Pamela Finkel
Lisa Holt
Lori Holzer
Debby Shishkowsky
Audree Smith
Vicky Svoboda
Ann Westine

Business Office
Steve Wagner
Kathleen Fry
Maureen Jester
Julie Kcomt
Gina Maschmeier
Rolanda Shelby
Larry Ball
Lee Dunayer
Michele Moretti
Robert Smith

Development Laboratory
Bob Parker

Electronic
Jeff LaCoss
Lee Magnone
Rick Shiffman
Jerry Wills

Mechanical
Shorty Garza
Bob Hines
Herb Yallop

Graphics
Curtis Nishiyama

Library
Sally Hambridge
Alicia Sabatine

Publications
Jim Melancon
Sheila Gardner

Computing Facilities Training and Information
Chloe Holg
Lisa Moses

Technical Support

Information Processing Centers

Dan Lynch, Director
Gary McGreal, Deputy Director
Dale Russell, Technical Assistant

Systems Development
Dennis Smith
- Dale Chase
- Helaine Fehling
- David Kompel
- Bill Moore
- Tom Wisniewski

Hardware
Dan Pederson
- Charlie Brown
- Glenn Gauthier
- Ray Mason
- Dan Trentham
- Leo Yamanaka

User Services and Administration
Jayne Millar
- Vicki Gordon
- Manon Levenberg
- Linda Sato
- Robin Swick

Software Development
Lynne Sims
- Mark Feber
- Ellen Lohneiss
- Eileen Machida
- Craig Ward
- Debbie Williams

Operations
Jim Hurd
- **Operators**
- Ron Shestokes
  - Stephen Carroll
  - Sam Delatorre
  - Walter Edmison
  - Brent Gesch
  - Norman Jalbert
  - Carlos Jimenez
  - Dwayne Tanner
  - Mike Zonfrillo

San Diego Facility
Serge Polevitzky
- Patrick Saunders
- Scott Smith

FUNDING SPONSORS

Air Force Office of Scientific Research
Defense Advanced Research Projects Agency -- Information Processing Techniques Office
National Science Foundation
Rome Air Development Center

PUBLICATIONS 1980 - 1981

Alter, R., and **J. A. Barnett,** "A postage stamp problem," *The American Mathematical Monthly* 87, (3), March 1980, 206-210.

Balzer, R. M., L. D. Erman, P. E. London, and **C. Williams,** "Hearsay-III: A domain-independent framework for expert systems," in *Proceedings of the First Annual National Conference on Artificial Intellegence,* pp. 108-111, Stanford, Calif., August 1980.

Balzer, R. M., and **N. M. Goldman,** "Principles of good software specification and their implications for specification languages," in *Proceedings of the National Computer Conference,* pp. 393-400, 1981.

Balzer, R. M., "Dynamic system specification," in *Proceedings of the Workshop on Data Abstraction, Databases, and Conceptual Modelling,* pp. 95-97, ACM, 1981. (Issued as a joint publication of SIGART, no. 74, January 1981; SIGPLAN, 16, (1), January 1981; and SIGMOD, 11, (2), February 1981.)

Balzer, R. M., "Transformational implementation: An example," *IEEE Transactions on Software Engineering* SE-7, (1), January 1981, 3-14.

Balzer, R. M., A. Cooperband, M. S. Feather, P. E. London, and **D. S. Wile,** "Application downloading," in *Proceedings of the IEEE Fifth International Software Engineering Conference,* pp. 450-460, San Diego, Calif., March 1981.

Barnett, J. A., M. I. Bernstein, R. A. Gillmann, and **I. M. Kameny,** "SDC research on speech understanding," in W. A. Lea (ed.), *Trends in Speech Recognition,* chapter 12, Prentice-Hall, 1980.

Barnett, J. A., "Computational methods for a mathematical theory of evidence," in *Proceedings of the Seventh International Joint Conference on Artificial Intelligence,* pp. 868-875, Vancouver, B.C., August 1981.

Barnett, J. A.: See Alter and Barnett.

Bochmann, G., and **C. Sunshine,** "Formal methods in communication protocol design," *IEEE Transactions on Communication* COM-28, (4), April 1980, 624-631.

Chiu, W. Y., "Structure comparison and semantic interpretation of differences," in *Proceedings of the First Annual National Conference on Artificial Intelligence,* pp. 259-262, Palo Alto, Calif., August 1980.

Cohen, Danny, and **J. Postel,** "On protocol multiplexing," in K. Thurber (ed.), *Tutorial: A Pragmatic View of Distributed Processing Systems,* pp. 437-443, IEEE Computer Society, May 1980.

Cohen, Danny, "Flow control for real-time communication," *ACM Computer Communication Review* 10, (1/2), January/April 1980, 41-47.

Cohen, Danny, "On packet speech communication," in *Proceedings of the Fifth International Conference on Computer Communication,* pp. 271-274, ICCC-80, Atlanta, Ga., October 1980.

Cohen, Danny, "DATACOM--the modern way," *Datamation* 27, (8), August 1981, 177-178.

Cohen, Danny, "The servant's dilemma," *Datamation* 27, (12), November 1981, 241.

Cohen, Danny, "A voice message system," in R. Uhlig (ed.), *Computer Message Systems,* pp. 17-28, North-Holland, 1981.

Cohen, Danny, "Packet communication of online speech," in *Proceedings of the National Computer Conference,* pp. 169-176, AFIPS, Chicago, Ill., May 1981.

Cohen, Danny, and **L. Johnsson,** "Computational arrays for the Discrete Fourier Transform," in *Proceedings of Compcon Spring '81,* pp. 236-244, February 1981.

Cohen, Danny, and **L. Johnsson,** "A VLSI approach to real-time computation problems," in *Proceedings of the Society of Photo-Optical Instrumentation Engineers: Real Time Signal Processing,* August 1981.

Cohen, Danny, and **L. Johnsson,** "A mathematical approach to modelling the flow of data and control in computational networks," in H. T. Kung, R. Sproull, and G. Steele (eds.), *VLSI Systems and Computations, Proceedings of the CMU Conference,* pp. 213-225, Computer Science Press, October 1981.

Cohen, Danny, "The VLSI approach to computational complexity (by Professor J. Finnegan)," in H. T. Kung, R. Sproull, and G. Steele (eds.), *VLSI Systems and Computations, Proceedings of the CMU Conference,* pp. 124-125, Computer Science Press, October 1981.

Cohen, Danny: See Karplus and Cohen; Postel et al.; Shoch et al.

Cooperband, A.: See Balzer et al.

Crocker, S. D., L. Marcus, and **D. van-Mierop,** "The ISI microcode verification system," in G. Chroust and J. P. Muhlbacher (eds.), *Proceedings of the IFIP Working Conference on Firmware, Microprogramming, and Restructurable Hardware,* pp. 89-103, North-Holland, 1980.

Crocker, S. D.: See Overman et al.

DiCiccio, V., J. Field, E. Manning, and **C. Sunshine,** "Alternatives for interconnection of public packet switching data networks," in K. Thurber (ed.), *Tutorial: A Pragmatic View of Distributed Processing Systems,* pp. 376-381, IEEE Computer Society, May 1980.

Doyle, J., and **P. E. London**, "A selected descriptor-indexed bibliography to the literature on belief revision," *SIGART Newsletter*, (71), April 1980, 7-23.

Erickson, R. W., and **D. R. Musser**, "The *Affirm* theorem prover: Proof forests and management of large proofs," in W. Bibel and R. Kowalski (eds.), *Lecture Notes on Computer Science*, Volume 87: *Fifth Conference on Automated Deduction*, pp. 220-231, Springer-Verlag, 1980.

Erickson, R. W.: See Lee et al.

Erman, L. D., and **V. R. Lesser**, "The Hearsay-II speech-understanding system: A tutorial," in W. A. Lea (ed.), *Trends in Speech Recognition*, chapter 16, Prentice-Hall, 1980.

Erman, L.D., **F. Hayes-Roth**, **V. R. Lesser**, and **D. R. Reddy**, "The Hearsay-II speech-understanding system: Integrating knowledge to resolve uncertainty," *Computing Surveys* 12, (2), June 1980, 213-253. (Reprinted in B. L. Webber and N. Nilsson (eds.), *Readings in Artificial Intelligence*, pp. 349-389, Tioga, Palo Alto, Calif., 1981.)

Erman, L. D., **P. E. London**, and **S. F. Fickas**, "The design and an example use of Hearsay-III," in *Proceedings of the Seventh International Joint Conference on Artificial Intelligence*, pp. 409-415, Vancouver, B. C., August 1981.

Erman, L. D.: See Balzer et al.; Lesser and Erman; Smith and Erman.

Feather, M. S., "Some contrasts and considerations of an approach to modelling," in *Proceedings of the Workshop on Data Abstraction, Databases, and Conceptual Modelling*, pp. 121-123, ACM, 1981. (Issued as a joint publication of SIGART, no. 74, January 1981; SIGPLAN, 16, (1), January 1981; and SIGMOD, 11, (2), February 1981.)

Feather, M. S.: See Balzer et al.

Fickas, S., "Automatic goal-directed program transformation," in *Proceedings of the First Annual National Conference on Artificial Intelligence*, pp. 68-70, Palo Alto, Calif., August 1980.

Fickas, S. F.: See Erman et al.

Gerhart, S. L., "Fundamental concepts of program verification," in *Proceedings of American Society of Mechanical Engineers, International Computer Technology Conference*, pp. 305-314, San Francisco, Calif., August 1980.

Gerhart, S. L., et al., "An overview of *Affirm*: A specification and verification system," in *Information Processing 80*, pp. 343-347, IFIP, Melbourne, Australia, October 1980.

Gerhart, S. L.: See Lee et al.

Gillmann, R. A.: See Barnett et al.

Goldman, N. M.: See Balzer and Goldman.

Hayes-Roth, F., **P. Klahr**, and **D. J. Mostow**, "Advice taking and knowledge refinement: An iterative view of skill acquisition," in J. A. Anderson (ed.), *Cognitive Skills and Their Acquisition*, pp. 213-253, Erlbaum, 1981.

Kameny, I. M.: See Barnett et al.

Karplus, W., and **Danny Cohen**, "Architectural and software issues in the design and application of peripheral array processors," *IEEE Computer* 14, (9), September 1981, 11-17.

Kini, V.: See Overman et al.

Klein, S., and **S. Sastry**, "Parameterized modules and interconnections in unified hardware descriptions," in M. Breuer and R. Hartenstein (eds.), *Computer Hardware Description Languages and their Applications*, pp. 185-195, North-Holland, 1981. (Proceedings of the Fifth International Conference, Kaiserlautern University, September 1981.)

Lee, S. L., **R. W. Erickson**, and **S. L. Gerhart**, "Finding a design error in a distributed system: A case study," in *Proceedings of the IEEE Symposium on Reliability in Distributed Software and Database Systems*, pp. 61-69, July 1981.

Lesser, V. R., and **L. D. Erman**, "Distributed interpretation: A model and experiment," *IEEE Transactions on Computers* C-29, (12), December 1980, 1144-1163.

Lingard, R., "A software methodology for building interactive tools," in *Proceedings of the Fifth International Conference on Software Engineering*, pp. 394-399, March 1981.

Lingard, R.: See Wilczynski et al.

London, P. E.: See Balzer et al.; Doyle and London; Erman et al.

London, R. L., and **L. Robinson**, "The role of verification techniques and tools," in W. E. Riddle and R. E. Fairley (eds.), *Software Development Tools*, pp. 206-212, Springer-Verlag, 1980.

London, R. L., **W. A. Wulf**, and **M. Shaw**, "Achieving quality software: Reflections on the aims and objectives of Alphard," in A. I. Wasserman (ed.), *Tutorial: Programming Language Design*, pp. 29-37, IEEE, 1980.

London, R. L., W. A. Wulf, and **M. Shaw,** "Abstraction and verification in Alphard: Defining and specifying iteration and generators," in A. I. Wasserman (ed.), *Tutorial: Programming Language Design,* pp. 145-155, IEEE, 1980. (Originally appeared in *Communications of the ACM,* August 1977.)

Mann, W. C., and **J. A. Moore,** "Computer generation of multiparagraph English text," *American Journal of Computational Linguistics* 7, (1), January-March 1981, 17-29.

Mann, W. C., "Two discourse generators," in *Proceedings of the Conference, 19th Annual Meeting of the Association for Computational Linguistics,* pp. 43-47, Sperry Univac, 1981.

Mann, W. C., "Selective planning of interface evaluations," in *Proceedings of the Conference, 19th Annual Meeting of the Association for Computational Linguistics,* pp. 33-34, Sperry Univac, 1981.

Mann, W. C.: See Moore and Mann.

Marcus, L., and **H. Samet,** "Purging in an equality data base," *Information Processing Letters* 10, 1980, 89-95.

Marcus, L., "The number of countable models of a theory of one unary function," *Fundamenta Mathematicae* CVIII, 1980, 171-181.

Marcus, L.: See Crocker et al.

Mark, W., "Rule-based inference in large knowledge bases," in *Proceedings of the First Annual National Conference on Artificial Intelligence,* pp. 375-381, Palo Alto, Calif., August 1980.

Mark, W., "Use of database organization in the Consul system," in *Proceedings of the Workshop on Data Abstraction, Databases, and Conceptual Modelling,* pp. 155-157, ACM, 1981. (Issued as a joint publication of SIGART, no. 74, January 1981; SIGPLAN, 16, (1), January 1981, and SIGMOD, 11, (2), February 1981.)

Mark, W., "Representation and inference in the Consul system," in *Proceedings of the Seventh International Joint Conference on Artificial Intelligence,* pp. 375-381, Vancouver, B.C., August 1981.

Matthiessen, C. M. I. M., "A grammar and a lexicon for a text-production system," in *Proceedings of the Conference, 19th Annual Meeting of the Association for Computational Linguistics,* pp. 49-55, Sperry Univac, 1981.

Moore, J. A., and **W. C. Mann,** "A snapshot of KDS, a knowledge delivery system," in *Proceedings of the Conference, 17th Annual Meeting of the Association for Computational Linguistics,* pp. 51-52, August 1979.

Moore, J. A.: See Mann and Moore.

Musser, D. R., "Abstract data type specification in the ***Affirm*** system," *IEEE Transactions on Software Engineering* SE-6, (1), January 1980, 24-32.

Musser, D. R., "On proving inductive properties of abstract data types," in *Proceedings of the Seventh ACM Symposium on Principles of Programming Languages,* pp. 154-162, ACM SIGPLAN, January 1980.

Musser, D. R.: See Erickson and Musser.

Overman, W. T., S. D. Crocker, and **V. Kini,** "A multiprocessor description language," in *Proceedings of the National Computer Conference,* pp. 19-25, AFIPS, 1981.

Postel, J., "DoD Standard Internet Protocol," *ACM Computer Communication Review* 10, (4), October 1980, 12-51.

Postel, J., "DoD Standard Transmission Control Protocol," *ACM Computer Communication Review* 10, (4), October 1980, 52-132.

Postel, J., "Internetwork protocol approaches," *IEEE Transactions on Communication* COM-28, (4), April 1980, 604-611. (Also in K. Thurber (ed.), *Tutorial: Office Automation Systems,* pp. 112-119, IEEE, September 1980.)

Postel, J., C. Sunshine, and **Danny Cohen,** "The ARPA internet protocol," *Computer Networks* 5, (4), July 1981, 261-271.

Postel, J.: See Cohen and Postel.

Sastry, S.: See Klein and Sastry.

Schwabe, D., "Formal specification and verification of a connection-establishment protocol," in *Proceedings of the Seventh Data Communications Symposium,* pp. 11-26, ACM/IEEE, October 1981.

Shoch, J., Danny Cohen, and **E. Taft,** "Mutual encapsulation of internetwork protocols," *Computer Networks* 5, (4), July 1981, 287-301.

Smith, A. R., and **L. D. Erman,** "NOAH: A bottom-up word hypothesizer for large-vocabulary speech-understanding systems," *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-3, (1), January 1981, 41-51.

Stotz, R.: See Wilczynski et al.

Sunshine, C., "Current trends in computer network interconnection," in T. A. Rullo (ed.), *Advances in Data Communication Management,* pp. 147-154, Heyden, 1980.

Sunshine, C., "Formal modeling of communication protocols," in *Proceedings of the Conference on Communication in Distributed Data Processing Systems,* Technical University Berlin, 1981.

Sunshine, C., "Local network protocols," in *Proceedings of the Local Networks and Distributed Office Systems Conference*, pp. 245-261, Online Conferences, Ltd., London, May 1981.

Sunshine, C., "Protocol workshop report," *ACM Computer Communication Review* 11, (3), July 1981, 7-8.

Sunshine, C. (ed.), *Communication Protocol Modeling*, ARTECH House, Dedham, Mass., 1981.

Sunshine, C., "Transport protocols for computer networks," in F. Kuo (ed.), *Protocols and Techniques for Data Communication Networks*, pp. 35-77, Prentice-Hall, 1981.

Sunshine, C.: See DiCiccio et al.; Bochmann and Sunshine, Postel et al.

Swartout, W. R., "Explaining and justifying expert consulting programs," in *Proceedings of the Seventh International Joint Conference on Artificial Intelligence*, pp. 815-822, August 1981.

Tugender, R.: See Wilczynski et al.

van-Mierop, D.: See Crocker et al.

Wilczynski, D., R. Stotz, R. Tugender, and **R. Lingard,** "Message system architecture--Experience at CINCPAC with the SIGMA system," in *Proceedings of Compcon Spring '80*, pp. 170-173, February 1980.

Wilczynski, D., "Consul: An experiment in building a cooperative interactive system," in *SHARE Proceedings*, pp. 8-12, March 1981.

Wilczynski, D., "Knowledge acquisition in the Consul system," in *Proceedings of the Seventh International Joint Conference on Artificial Intelligence*, pp. 135-140, Vancouver, B.C., August 1981.

Wile, D. S., "Type transformations," *IEEE Transactions on Software Engineering* SE-7, (1), January 1981, 32-39.

Wile, D. S.: See Balzer et al.

Williams, C.: See Balzer et al.

USC/Information Sciences Institute
4676 Admiralty Way, Marina del Rey, California 90291
1982