SIGDA NEWSLETTER

SPECIAL INTEREST GROUP ON DESIGN AUTOMATION

Volume 5 Number 4 December 1975

Contents:

SIGDA

ACM Special Interest Group on Design Automation

ADDRESSES

CHAIRMAN:
Charles W. Rose
Computing & Information Science
Case Western Reserve University
Cleveland, Ohio 44106
(216) 368-2800

VICE-CHAIRMAN:
Judith G. Brinsfield
Bell Laboratories
Building 3B-323
Whippany, NY 07981
(201) 386-3169

SECRETARY-TREASURER:
Luther C. Abel
Digital Equipment Corporation
146 Main Street
Maynard, Mass. 01754
(617) 897-5111

EDITOR:
Robert J. Smith, II
Lawrence Livermore Laboratory, L-156
P.O. Box 808
Livermore, CA 94550
(415) 447-1100, X-8088

TECHNICAL COMMITTEE:
Judith G. Brinsfield

MEMBERSHIP COMMITTEE:
William M. vanCleemput
Digital Systems Laboratory
Stanford University
Stanford, CA 94305
(415) 497-1270

PUBLICITY COMMITTEE:
Lorna Capodanno, Chairman
Bell Labs
2C169
Murray Hill, New Jersey 07974
(201) 582-6909

BOARD OF DIRECTORS:
Dr. John Grason
Department of Electrical Engineering
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213

Dr. Edwin Hassler
Design Automation Department
Components Group
Texas Instruments, Inc.
P.O. Box 5012
Dallas, Texas 75222

Professor James Linders
Department of Applied Analysis
University of Waterloo
Waterloo, Ontario
Canada

BOARD OF DIRECTORS, cont.'d
Dr. Ralph Miller
Structures Research
Commercial Airplane Group
Boeing Company
P.O. Box 3707
Seattle, Washington 98124

Dr. Jean-Marie Comeau
Bureau of Management Consulting
365, Ave Laurier Ouest
Ottowa, Canada
K1AOT5

MEMBERSHIP

SIGDA dues are $3.00 for ACM members and $5.00 for non-ACM members. Checks should be made payable to the ACM and may be mailed to the SIGDA Secretary/Treasurer listed at left, or to SIGDA, ACM Headquarters, 1133 Avenue of Americas, New York, N.Y. 10036. Please enclose your preferred mailing address and ACM Number (if ACM member).

SIG/SIC ACTIVITIES

1) Informal technical meetings at NCC.
2) Formal meeting during National ACM meeting + DA Workshop.
3) Joint sponsorship of annual Design Automation Workshop.
4) Quarterly newsletter.
5) Panel and/or technical sessions at other National meetings.

FIELD OF INTEREST OF SIGDA MEMBERS

Theoretic, analytic, and heuristic methods for:

1) performing design tasks,
2) assisting in design tasks,
3) optimizing designs through the use of computer techniques, algorithms and programs to:

   1) facilitate communications between designers and design tasks,
   2) provide design documentation,
   3) evaluate design through simulation,
   4) control manufacturing processes.

CHAIRMAN'S MESSAGE

As you can see from the listing on the inside cover, we have made a lot of progress in appointing committee chairmen and a Board of Directors. I am still waiting for responses from two persons whom I asked to serve.

The International Symposium on Computer Hardware Descriptive Languages and the Data Base Workshop which we sponsored in September were, by all reports, quite successful. A report on the Data Base Workshop appears elsewhere in this issue. I would like to thank Jim Linders, Bill vanCleemput, Chuck Radke, and the other members of SIGDA who worked so hard to make these events the successes that they were.

During ACM '75, the DIG/SIC chairpersons met. While a wide range of subjects were discussed, those most relevant to SIGDA were: (1) headquarters charges for non-ACM SIG members and for subscribers will be raised significantly beginning 1 July 1976. Between now and then, your officers, after consultation with the Board of Directors, will set a new dues structure for these categories of SIGDA affiliates; (2) referees and papers are needed for ACM '76 in Houston, and SIG's have been asked to furnish lists. I'll have more to say on that below.

SIGDA is now part of a Special Interest Area on Technology together with SIGARCH, SIGGRAPH, SIGCOMM, SIGMICRO, and SIGMINI. This SIA hopes to function actively in forming a yearly area publication, sponsoring joint technical sessions at ACM Conferences, and sponsoring interface workshops among members.

I have agreed, with John Tartar, chairperson of SIGMINI to organize a joint technical session at ACM '76 on Minicomputers in Design Automation. The session will focus on distributed intelligence in DA systems, standalone mini-based DA systems, and architectural features on minicomputers affecting their suitability for DA applications. There is a "call for papers" elsewhere in this issue.

OUTGOING CHAIRMAN'S MESSAGE

As outgoing Chairman I wish the new officers the best as they and the organization continue to strive toward the SIGDA objectives. To those, especially Dave Hightower, Lori Capodanno, and Rob Smith, who have helped me for the last two years, my heartfelt thanks.

You know that the accomplishments within the area of Design Automation are not always easy to see. A clear measure of the productivity gains and design aids are not usually available.

A machine may be made to stamp out a thousand identical parts. The productivity gains are quantitatively clear. Not too many companies or even individuals desire to make each of the thousand parts slightly different. Ah, but now comes the rub! The products of design automation are all different. Each design processed or aided by your system <u>must</u> be different in some way. One manufactures identical parts, one does not design identical parts.

Of course, as soon as one aid is provided it is used beyond its limits and someone immediately says "hey it doesn't work" or "give me a pencil and paper. I can do it better myself."

The problems facing us are endless. It is nice to know that one never lacks for a challenge (although not all challenges are to our liking). Sometimes we conquer a challenge and no one (we believe) realizes our accomplishment.

I found it interesting to discover in several conversations at the Boston conference the same frustrations - and the eagerness to meet headon an ever continuing set of new challenges.

Over a cup of coffee at a conference, such as the one in Boston or the ones coming up in Waterloo and New York, we get that new idea to meet those challenges. Maybe during the writing of a paper and after reading the referee's comments we convert our frustrations into progress. Sometimes we even catch a glimpse of the value of a professional organization like ACM and subgroups as SIGDA.

As I turn the riens of SIGDA over to Chuck Rose, I look at the satisfaction I have received. I think of the support which I have received from many of you. I remember the discussions which we have had, the papers I have reviewed, the sessions I have chaired.

As I told Chuck Rose, I'll be around, give me a call! I know you'll be there too.

Chuck Radke
SIGDA Chairman
January 1971 - July 1975

FROM THE EDITOR

I would like to recognize the contributions arranged for this issue by Prof. John Gero of the University of Sidney, Dept. of Architectural Science, who has volunteered to serve as Associate Editor for Architectural Design Automation.

Thanks also to Bill vanCleemput, Jim Linders, and Steve Su for their contributions!

Please send me materials for the next issue no later than February 1, 1976.

Rob

"<u>THE EASY WAY</u>. Planning a conference? Symposium? Technical Meeting? Worried about conflicts of proposed dates with meetings of other SIGS? Call Smith Dorsey of the Conferences and Symposia Committee, 714-632-1391, to check possible conflicts and to establish your date."

CONTROL OF DESIGN DATA

IN THE

INTEGRATED SHIP DESIGN SYSTEM

Peter R. Bono
Naval Ship Engineering Center

KEYWORDS: computerized design system; computer-aided design; design data integrity; design file control

Abstract

The Navy's Integrated Ship Design System (ISDS) is being designed as a collection of application program modules (for preliminary design) which communicate with a centralized set of data files. These files use the existing COMRADE Data Management System which was designed specifically for integrated systems.

Apart from providing an environment in which to operate the engineering application modules, ISDS's main role is to manage the creation, flow and archiving of the ship design data and to control access to this data. Consequently, a major concern during the lengthy and complex ship design process is assuring the integrity of the design data as it grows and is revised over time.

Planning for control of the design data requires a clear understanding of the design process and the interrelationships between the design tasks. Requirements are stated, problem areas are identified, and possible approaches for implementation are suggested.

Background

The U.S. Navy has undertaken a multi-year, multi-million dollar Computer-Aided Ship Design and Construction (CASDAC) project (1). The first implementation phase includes the development of an Integrated Ship Design System (ISDS) (2)--a collection of application programs for the preliminary design of ships which communicates with a centralized set of data files using a standard language provided by the COMRADE data base management system (3,4,5,6,7).

One goal of ISDS is to increase and facilitate communication between separate engineering programs written to solve specific portions of the total ship design process. Many items of data are common to a spectrum of analysis programs and ought to be maintained in a common data base so that all programs will use current information. Planning for the proper flow of data between the data base and the diverse collection of application programs is described by the functional requirements to be met by ISDS and by the problems encountered when trying to satisfy these requirements.

This paper will pose more questions than answers and describe more problems than solutions regarding how to properly manage design data in a complex system. It is hoped that the description will help those who are about to embark upon such a computer-aided design system and, conversely, will elicit suggestions from those who are further along than we, so that we may, in turn, benefit from the experiences they have accumulated.[1]

The Data Base

As presently constituted, the data base for ISDS consists of a special file called the Ship Design File (SDF) and a collection of catalog files. The distinction between the ship design file and catalog files is somewhat arbitrary; but, for the moment, it has been valuable to consider the SDF as containing those elements which are specific to a particular ship design. Catalog files, conversely, contain information which apply to many different types of ships. For the rest of the paper, we will describe only the control of data on the ship design file since this data is changing most often and elucidates most of our problems, especially in the early stages of design.

The ship design file contains different types of data: geometric values (both coordinate and pictorial), parametric information, fixed and floating point numbers, and alphanumeric strings. In the later stages of design, an individual SDF will require two to three million words of mass storage.

Functional Requirements

The potpourri of problems relating to controlling design data has been partitioned into a number of areas (see Table 1). The partitioning is rather arbitrary since each problem may impact

[1]Information has been provided to researchers working on NASA's Integrated Program for Aerospace-Vehicle Design (IPAD) Project. We have recently reviewed the final report concerning IPAD feasibility (8,9).

TABLE 1

Problems in the Control of Design Data

- STATIC
  - File Structure
    - logical
    - physical
  - File Integrity
    - accuracy
    - consistency
  - Data Dictionary
    - read/write permission
    - authoritative definition
    - format specification
- DYNAMIC
  - Lower Level
    - update
      - empty slot
      - change previous value
    - notification
    - validation
    - groups of elements
    - merge/extract
  - Higher Level
    - levels of confidence
    - use of module
      - integrated system
      - stand-alone
    - restart/design
      - planned iterations
      - expected minor changes
      - traumatic major changes
- ADMINISTRATIVE
  - Reports
    - status
    - critical events
  - Design Process Model
    - sequencing
    - status
  - Design History
    - archival record
    - logistical support
    - reference for future

directly, or at least broadly influence, problems in other areas. It is designed primarily as an intellectual device to make the discussion more coherent.

Each problem is introduced by stating a requirement that ISDS must satisfy in order to be a viable integrated system. Some problems have been discussed in the literature for many years and are included here for completeness only; they are given minimal attention.

Static Considerations

Each ISDS application module requires that the data base be structured to serve its own needs. Yet these needs differ widely, so ISDS must provide a means for multiple logical file structures to be mapped into the single physical file structure selected for the Ship Design File. For example, for some programs it is appropriate to think of the data as being structured as a hierarchical tree structure with no cross links, etc. Other programs might prefer to see portions of the data as being structured as a double-linked list or as an indexed-sequential file. To accommodate all these different views of the data in a single physical file structure usually involves determining which logical structure is the one that will be used most frequently and having the physical structure of the file reflect this desired structure. The remaining views of the file or logical structure of the file will then have to be simulated or provided via indirect means like extra pointer elements, flags, etc.

Another static consideration is that ISDS is required to maintain file integrity. There are two components to file integrity. The first relates to the accuracy of the data in the file. By "accuracy" one means an absolute confidence that the numerical value of the data element is the desired one. There is also another concept, to be called consistency. This reflects relative confidence in the data. It is not that the number is necessarily correct but that, given the total collection of data values on the file, they are consistent among themselves. For example, if C is a function of A and B, and if A and B have certain values, then indeed in the location labeled C one should find a value that is equal to the desired function of A and B. This relative confidence should not be confused with absolute confidence because, in the example just cited, it is perfectly possible that if A and B, the input parameters or the independent variables, are inaccurate, then, of course, C, the dependent variable, will also be inaccurate in an absolute sense. Nevertheless, the data will still be consistent. Problems with inconsistency arise when A or B is changed and C is not updated.

ISDS is required to permit responsibility (read/write/modify control) for data elements to be exercised by more than one application program. Ideally, it would be best if only one program were permitted to write to each data element on the file. In that way each data element could be associated with a program which had the responsibility for it and all changes to the file would be made by executing that single program. Thus, one would have better control over the consistency of the data in the file. In many circumstances, this is not possible: there are overlapping responsibilities. Usually this results from having the same piece of data being used in different stages of the design process. For example, in the early stages of design, values for the total manning of a ship may be provided and, later on, a different group of people, at a much more detailed stage, estimates this number. If that data is stored in the same slot, then there is a conflict because both design groups have the ability to alter this value and if one group alters it without letting the other group know, the consistency of the file is jeopardized.

Dynamic Considerations

During the design cycle, each write to the SDF by an application program requires ISDS to monitor the integrity of the file and to notify the appropriate people of changes made to the file.

The basic problem of updating an element, on the surface, appears rather trivial; yet even at this point there are problems because one must distinguish between filling an empty slot (that is, there was a data element created but no numerical value was assigned to it) and changing a previous value assigned to a data element. Procedures to be followed to assure a consistent file are different in the two cases. Namely, if one changes a previous value, one must ascertain that the program and person doing the changing has the authority to do so and that making the change will not make the file inconsistent if this particular item happens to be a function of other parameters which have not been altered.

This introduces another problem; namely that of validation, which is really a series of consistency checks to be carried out at update time. A major problem that arises in ship design with so many design passes is illustrated in the following: Input data is retrieved from the ship design file at a certain point in time and copied to local working files for use by the application programs. At some later time--hours, days, or even weeks later--the application designer has executed his program many times, has come up with a design which meets his criteria, and wants to update the ship design file. At this point, before his output or derived data can be permitted to be stored on the file, it must be ascertained whether the input data on which he based his design has been changed or not. If it has been changed, then there must be initiated a notification procedure which determines whether this update should be allowed, if it is allowed, who should be notified about the change in the data, and what that notification should consist of.

Even notification is not all that simple: multiple levels of notification are often desired. There are general changes to the ship design file which one would wish to pass on to the design manager's staff on periodic basis (routine status reports reflecting the status of the design at any particular time). One must also have the ability to reflect specific changes of specific data elements to design specialists. For example, if a certain pump is changed to a higher-capacity pump, then the mechanical, electrical, and structural groups need to be informed so that they can change their total power estimates, their cooling estimates, arrangements, etc. There is really a continuum of change notifications that need to be made, and economic cost/benefit trade-off studies would have to be done to determine what is an appropriate level for a particular design.

Even if an update will maintain the consistency of the file, validation of accuracy is still a concern. That is, if an application program module malfunctions, it should be determined that the data just stored on the file is reliable to the extent that the computer can ascertain. Mechanisms proposed involve limits, e.g., knowing that one cannot have a negative speed or shaft horsepower. There are more sophisticated limit checks that one can imagine which would involve checking the value of a particular element and comparing it as a function of other variables that are already stored on the data file.

All these issues have been dealt with at the simple data element level. There are also higher levels of change. That is, in our data management system, we have blocks, subblocks, repeating groups, arrays, etc. Sometimes changes are being made to whole portions of the file rather than an element at a time. For no other reason than efficiency, it would be more desirable to make modifications and validity checks at high levels than on an element-by-element basis.

When a basic operation, from the user/engineer point of view, is executed, ISDS is required to determine all of the indirect effects of making the change and to take appropriate action. For example, if equipment B replaces equipment A in compartment C, then forward and back pointers between the equipment descriptions and the description of compartment C must be added (for B) and deleted (for A). In addition, if lists of operational ship requirements are maintained, these lists must be modified to reflect the substitution of equipment. For example, the electrical cable routing information may have changed, the ventilation requirements for that particular compartment may have changed, etc. There is an extremely extensive and complex network of interrelationships among all elements on a ship. These may never be adequately managed by a computer, but they are real problems that must be solved by some combination of man and computer.

ISDS must provide the ship designer with information concerning the reliability of the input data he is about to use for his portion of the effort. It is a first-pass guess, a calculation based on the execution of previous programs, etc? Reflecting these levels of confidence in the accuracy of the data values on the ship design file becomes particularly imperative because the ship design process is not a purely serial process. If it were, a ship would never get designed. That is, the people who are concerned about air conditioning or fuel capacity must begin to make estimates and must begin to arrange their equipment into ship compartments before completely accurate estimates as to the weapon systems, the manning, etc. have been determined. So they must start with guesses and then work in parallel with people who are trying to refine other portions of the ship design. These guesses are replaced with more refined and precise values with each design pass. These problems within the design process are not new; however, the integration of the computer requires formalized statements of problems previously solved on an ad hoc basis through personal relationships.

Another aspect of higher level integration is just the simple use of the module. At NAVSEC, programs need to be used both within an integrated system context and as stand-alone modules in which the individual user/engineer is permitted to provide input data over which he usually has no cognizance or responsibility. There are two approaches one can take. One can have two versions of the program: one which runs under the integrated system and does not permit entering data which would violate the consistency of the ship design file, and another--a stand-alone version--which would, of course, permit this, but would then not be able to interact with the ship design file. The disadvantages of the two-version course are fairly obvious: the two versions of the program never remain the same regardless of how well intentioned one is and one ends up with one or the other version being enhanced to the point that the other one is unacceptable and will not be used. If one tries to accommodate the stand-alone and the integrated design needs within the same module, then one is very definitely jeopardizing the consistency of the data on the SDF and some fairly rigid controls, with their associated overhead, must be built into the single program which is expected to be run in both stand-alone and integrated modes.

At the program flow level, an integrated system must be able to handle three kinds of restarts: <u>planned</u> multiple design passes due to the interactive nature of the design process, <u>expected</u> minor restarts due to specified engineering changes, and <u>traumatic</u> major restarts of major systems. We would like to be prepared for these contingencies in terms of being able to retain these portions of the design that still remain valid during all these planned, expected, or traumatic design changes. At this level, we are beyond the powers of our comprehension of integrated systems, and a substantial amount of manual intervention will be required and indeed is desirable. As system designers we must permit and allow for this kind of manual intervention. It is not so much that we must make a system that automatically reacts to all contingencies, but rather we must put the hooks in, we must prepare to enhance our system as our knowledge increases and our budgets allow.

Administrative Uses of the Ship Design File

One of the most obvious uses of design data is for status reports. There is a vast amount of information that can be gathered from the ship design file and provided in reports to be issued to various people at various stages in the design process. These reports need to be generated both during the execution of application programs and by separate programs whose sole purpose it is to produce certain reports for managers, for technical people, etc.

Another high-level administrative function to be provided is a model of the design process in which the sequence of the design steps taken is monitored by the model. This would provide a means of determining the status of the design, how many iterations have taken place, how many different aspects of the total design have been considered, etc. This modeling of the design process would be extremely complex to accomplish because it is barely understood at present in the sense that the responsibilities for the total design of a ship are distributed over such a large number of people that determining exactly what the complete process is, is a task in itself.

An important aspect in integrated ship design is the concept of having a design history; that is, "snapshots" of the design at certain designated points, which document the progress of the design and serve as milestones for review by higher authority and by the design managers themselves. In addition, they may be able to serve as examples of designs which have been successful in the past and be used in future designs where similar functions must be provided. Also in this vein, at the end of a complete design, the ship design file serves as an archival record of the ship or the ship class which hopefully would be able to be used for logistical support, overhauls, and modifications which are to be made many years after the ship has gone into active service.

Finally, aspects of administrative control that we cannot ignore but are computer science problems rather than people or design problems are those of backup of the files, recovery from system and disk crashes, and restarts necessitated by temporary computer failures. We have an obligation to restore the designer close to the point at which he was when the system crashed in order to avoid his having to redo major portions of his analysis or data input or data verification. Although this problem has been recognized for many years, progress has been slow, despite its being one of the most fundamental factors determining whether our large systems will be accepted by production shops--engineers operating in time-critical, performance environments.

Approaches to Solving These Problems

This section will describe the approaches NAVSEC has taken in attacking some of the problems posed in the previous paragraphs. We are trying to build a system in an evolutionary manner, phase it in, have its portions used, get feedback from the designer/user, and modify our procedures according to these human engineering factors as we begin to accumulate information about them. Our approaches to some of our problems do not yet come close to "solving" the problems and many are not new or unique.

** We have set up a data dictionary for the ship design file which lists the element name, the format of the value that can be placed or associated with that element name, what its meaning is, whether it is a pointer, whether it is a number, how it is related to other elements in the file, etc. This data dictionary is under a single person's control and it parallels standard practices regarding the concept of a data administrator.

** We are working on a study of a data description language that would permit us to separate the logical data structure from the actual data values assigned along the lines of the CODASYL Report (10). Our efforts in this area are very tentative, and we may decide to abandon them if if is not cost-effective for us to pursue it. Nevertheless, there is a potential here for permitting different application programs to view the ship design file as having different logical data structures.[2] This increases the independence of the various application modules and allows a more efficient development of these applications programs.

** Our present view of how we are going to update a portion of the design onto the ship design file is as follows: Basically, the application program at some point in time extracts data from the SDF onto the working file and does its design or variety of designs. At some later time, a program freezes the ship design file and compares it against the selected working file design for consistency of input data extracted. If the input values have not changed, an update of the derived new data is permitted and the ship design file is then released for use. If consistency checks show that the selected working file design was based on data that have now been altered, then messages to that effect will be issued and the update will not be permitted to take place. The individual designer will then have to go back and determine what the changes are, whether they are significant to his design, and either redo his design or get the input parameters that he based his design on changed back to what he had originally expected them to be.

** Presently, we are barely attacking the level of confidence issue. We do have working files which have, in some sense, less precise information in them than does the "pure" ship design file. We have considered associating integer confidence indicators with items of data. But there are problems associated with doing this; for example, since the level of confidence of the output should be no higher than the lowest level of confidence of the input data items, the mathematics or pseudomathematics could get very involved and might be self-defeating. Yet we recognize a need for individuals to have some understanding, some sense of how valid the data on the ship design file is.

** The Computer-Aided Design Environment (COMRADE) software was developed in response to requirements which include accommodating multiple design passes and expected engineering changes. References 2 and 3 detail the functional requirements and the impelmentation strategy which led to a successful outcome. With COMRADE, we feel very comfortable in an iterative, interactive mode of operation. However, we have not started to cope with the traumatic restart/redesign problem.

** On the administrative side of things, we are working on the specifications for a report generator to be integrated into the COMRADE system. There are a number of ways for implementing a report generator: as a compiler producing an executable program, as an interactive interpreter accepting command lines, and as a number of intermediate forms. We have not fixed on any one form: we may have to supply all of them to meet the variety of

[2]The current data structure of the SDF is described in reference 11.

needs for reports. At present, COMRADE has a query capability on COMRADE files, but is not a general purpose report program and it is not easy to get large amounts of data printed out in a satisfactory format.

** In terms of backup, we have just started this past year to look at the problem. We provide a disk-to-tape, tape-to-disk backup cycle, grand-father-father-son in the standard, traditional manner. Its advantage is that it is simple; its disadvantages are that it is time-consuming and incapable of sophisticated manipulation. This in no way begins to attack the restart/recovery problem.

** We have lots of ideas about a design process model in which the levels of confidence, global status flags, and such esoteric concepts as validation subroutines, etc. would be built up in parallel with a ship design file which was "pure" in both an accuracy and a consistency sense. Control and monitoring of the design data would basically function through the design process model in a transaction-oriented scheme. We have not pursued these thoughts very far--partially because of available resources and partially because the need to solve these problems has not been compelling. We have enough time to consider a variety of alternatives. There may not be a "best" way to go: we may have to choose one way for preliminary design and modify it as the design becomes more detailed, as the process becomes more complex.

Conclusion

Controlling the design data is critical to an integrated design system. No matter how good your engineering algorithms, no matter how excellent your intuitive understanding of the processes involved, that will not be sufficient to insure a successful design system. There are too many people involved, too many things happening concurrently, too many vagaries of the computer system--hardware and software--to ignore any of the problems raised here. There are undoubtedly more problems that will surface; or rather, each of these problems will branch into a multitude of subproblems, each of which will need to be attacked. However, each problem cannot be attacked by itself without awareness and consideration of the interrelationships with the other problems. It is a mistake to feel that all one's problems will relate to computer hardware and software. A complete understanding of the design process and a feeling for the human interactions that take place daily is essential to success. Success will only be achievable if the system is developed in a phased, incremental manner with a large degree of interaction with the final users in a production environment.

The first part of our integrated ship design effort, from the system software point of view, has been merely to assemble the tools: a data management system, quite sophisticated; an executive, again quite sophisticated; and the beginnings of a design administration system. Our task now is to use these tools, to make them available to the application programmers in such a way that all the pieces will work together. At present we have about eight modules working through a central ship design file. We are presently undergoing a validation of this proto-ISDS, and the results are yet to be determined. This paper is intended to elicit comments from those of you have been through this--comments concerning problems omitted or conceptualizations of the process that appear to be at variance with your own experience. This paper should prompt those of you who have been given the responsibility for developing an integrated design system to reflect upon the human aspects of the problem and not just on the computer side of things. With unlimited resources of manpower and money, we can probably solve most or all of our computer problems. But the "solution" to a computer problem may in fact be an intolerable burden on the user of the system. There has to be an optimization of both human engineering and system design factors to have any hope of success. I am optimistic in the long run that this can be achieved, but it is going to take cooperation between those of use who are responsible for the design and need the tools to get it done and those of us who understand computers and their capabilities and limitations but who are not completely aware of the parallelism of effort, the iterations that take place, the need for proceeding with incomplete information, the need to accommodate change, the need to provide for multiple modes of operation--in short, the need really to control the data in the ways brought out in this paper.

REFERENCES

1. Shether, A., "CASDAC: Computer-Aided Ship Design and Construction," All Hands, January 1974.

2. Brainin, J., "Use of COMRADE in Engineering Design," 1973 AFIPS National Computer Conference, New York, June 1973.

3. Rhodes, T., "The Computer-Aided Design Environment Project (COMRADE)," AFIPS National Computer Conference, New York, June 1973.

4. Tinker, R., and I. Avrunin, "The COMRADE Executive System," AFIPS National Computer Conference, New York, June 1973.

5. Willner, S., Bandurski, A., Gorham, W., and M. Wallace, "COMRADE Data Management System," AFIPS National Computer Conference, New York, June 1973.

6. Bandurski, A., and M. Wallace, "COMRADE Data Management System--Storage and Retrieval Techniques," AFIPS National Computer Conference, New York, June 1973.

7. Chernick, M., "COMRADE Design Administration System," AFIPS National Computer Conference, New York, June 1973.

8. "Feasibility Study of an Integraded Program for Aerospace-Vehicle Design (IPAD)," prepared under NASA Contract NAS1-11441, The Boeing Commercial Airplane Company, August 30, 1973.

9. Report of the Data Base Task Group to the CODASYL Committee, April 1971, published by AFIPS Press, New York.

10. Thomas, B., "Plex Data Structure for Integrated Ship Design," AFIPS National Computer Conference, New York, June 1973.

ETHICS IN COMPUTER-AIDED DESIGN: A POLEMIC*

JOHN S. GERO

Department of Architectural Science
University of Sydney, Australia

formerly

Harkness Research Fellow
Department of Architecture
University of California, Berkeley

ABSTRACT

It is one of the difficulties of the scientific approach to problem solving that the variables need to be defined at the beginning of the investigation. This is further exacerbated when computer based techniques are applied because of the need to define variables explicitly. Current problem solving is directed to handling only well defined problems in which certain variables are assumed to be exogenous - the educational system schools people in methods for manipulating problems at this level. It appears that computer-aided design systems, in general, have not been able to incorporate any adequate value systems within them nor have they been able to provide a means of examining the problem in an ethos borader than the one defined at the outset. Both of these difficulties are considered within the ambit of ethics. It is suggested that subjective value systems can be easily incorporated with the use of interactive computing but that the 'ethics of the whole system' present a thornier problem.

INTRODUCTION

> The gods did not reveal, from the beginning,
> All things to use; but in the course of time,
> Through seeking, men find that which is better.
>
> But as for certain truth, no man has known it,
> Nor will he know it; neither of the gods,
> Nor yet of all the things of which I speak.
> And even if by chance he were to utter
> The final truth, he himself would not know it;
> For all is but a woven web of guesses.
>
> - XENOPHANES

It is not the intention of this paper to present either a treatise on ethics or on computer-aided design but rather to set forth some notions on the relation between the two. The aim in presenting them is to provoke discussion about an area, that those involved in developing the panoply of systems which can be classified under the umbrella of computer-aided design, tend to forget and neglect.

---

*Paper presented at the Symposium on Basic Questions of Design Theory, Columbia University, New York, May 1974.

The rapid development of computers and the parallel growth in the mathematical techniques which are grouped under 'systems analysis and design' have resulted in a symbiotic relationship between the two within the context being considered here. If one takes the idea that one of the fruitful uses of computers is to have them manipulate models, then it seems reasonable to inquire about these models. Models may be deemed to be of three types:

(i) descriptive
(ii) predictive
(iii) prescriptive

Computer manipulatible models tend to be predictive in nature when the system is thought of as performing a simulation of the situation being modelled and are sometimes represented as prescriptive when the system is thought of as performing an optimization based on a defined objective. It is argued that simulation models tell you what will happen for a given set of conditions whilst it is further argued that optimization tells you what conditions are needed if some future situation is to occur. Neither of these two statements is an adequate description.

Fundamental to systems analysis is the idea of a subsystem: "there is a tradition in Western thought that parts of the whole system can be studied and improved more or less in isolation from the rest of the system" (1). It is common to listen to learned papers at conferences which commence with statements limiting the scope of the problem being considered. And then, at the end of the paper, the author concludes that he has now solved the problem without putting it back into its original context; without testing the validity of defining certain variables as being endogenous and others exogenous. Ultimately all variables must be endogenous. It is one of the difficulties of the scientific approach to problem solving that the variables have to be defined at the commencement of an investigation. This is further exacerbated when one wishes to apply computer based techniques because of the need to be able to define variables explicitly. In conversation one does not do this, even in philosophical analysis that is not a necessary condition - "the common confusion that makes people think they cannot understand an idea unless then can define it, forgetting that ideas are defined by other ideas, which must be already understood if the definition is to convey any meaning" (2). It is pertinent to ask why we seem to be able to manipulate subsystems with a relatively high degree of success when they are well bounded but fail when they are not.

Before attempting to answer this question, examine a hierarchy of decision-making in design in descending order of difficulty:

(i) recognition of problem
(ii) definition of problem
(iii) solution of defined problem
(iv) implementation of solution

(This dissection is not meant to intimate that each of these can be isolated from the others). A large part of the education of architects and engineers is directed towards the solution of defined problems. The tools which have been developed as part of a problem-solver's kit comprise largely techniques for handling area (iii) above; this is more apparent in engineering education than in architectural education but the same applies there also. Design problems in structural engineering education are often presented in the form:

> Given a set of loading conditions and a structural type 'design' the members to satisfy specified criteria.

This is an elementary subsystem of structural engineering design. Less frequently, the bounds of the defined problem are expanded so that the design problem may be presented in the form:

> Given a set of loading conditions and a gap to be spanned 'design' a suitable structure to satisfy specified criteria.

It is hard to imagine an engineering school where exogenous variables as defined by the instructors would be open to conversion to endogenous variables by the student. In architecture schools, design problems are set up in a similar way with the significant difference that criteria against which solutions may be evaluated are not always specified.

Why is it that design education appears to be much more involved with the 'easier' area of solutions of defined problems within well expressed bounds? The reason can simply be expressed using the analogy of the apocryphal story of the man who loses his car keys during the evening and spends the night looking for them under the lamp-post not because that is where he lost them but because that is where the light is: as Goethe said "Light, more light". Problem recognition and problem definition are difficult to teach, therefore, they are not taught although the definition of a problem affects its solution, but this feedback loop is ignored.

COMPUTER-AIDED DESIGN

Computer-aided design systems are rarely set up to emulate the human designer, rather they tend to utilize techniques which require computational capabilities not exhibited by humans. But, at their current level of development, they are set up to perform the decision-making in the well defined, bounded design problem with specified criteria. This should not be surprising in light of the above discussion. The boundaries of the design problem in computer-aided design systems need to be fixed in some manner before the system can be operated. The dictum 'if it cannot be computed, it cannot be included', is followed with religious fervor although all such systems are often set up with numerous non-compatible assumptions which are rarely, if ever, examined once made. More often than not, these assumptions are not stated, either through a disregard of them or because they are not thought pertinent to the problem: "many mathematicians who lack sufficient ... training may jump to the erroneous conclusion that no assumption is needed of no assumption is stated". (3).

This appears to be the crux of one of the fundamental difficulties of computer-aided design: in order to manipulate the problem with a particular set of tools, the problem is so constrained that it allows no feedback to the ethos from which it was extracted. No automated computer-aided design system provides the opportunity for examining the context and hence it will likely fail to solve the problem within any broader context than that defined by the exogenous parameters to that particular system, even though those exogenous parameters are endogenous variables for a somewhat larger problem.

Added to this difficulty is that fact that we appear to have only inadequate tools to handle the problem in its broader context (4,5), which undoubtedly accounts for the reason why computer-aided design systems fail to attempt to handle them.

Well-defined problems generally involve some value judgments in their formulation (definition) and, in part, it is this that adds to the difficulty of developing suitable algorithmic approaches to assist in their manipulation: "It is a fallacy to disregard criteria which involve value judgments as unscientific under all circumstances. There is a place for such value judgments and a need for them though this may not fit the prejudices of the promoters of the quantitative method as the only true scientific method" (6). Obviously, value judgments are made by individuals and groups as a matter of course in everyday life and there are areas in economics which aim at providing external measures of value - however, this continues to remain a thorny problem (7,8).

It would appear from the current state-of-the-art (9) of computer-aided design based on systems analysis that it has not been able to provide a satisfactory method of incorporating an adequate value system within itself nor been able to provide a means of examining the problem in an ethos broader than the one defined at the outset of the problem solution phase. Attempts to do so within the strict notions of scientific systems analysis have failed, possibly because those techniques are either inappropriate or inadequately developed: "if the only tool you have is a hammer, it is tempting to treat everything as if it were a nail" (10). Should it be inferred from this that systems analysis and the scientific approach are not just inadequate but need to be thrown over?

ETHICS IN COMPUTER-AIDED DESIGN

Both of these objections fall within the ambit of ethics because when one talks about subjective values one needs to entertain the notions of good and bad. When one talks about a problem within its ethos one needs also to entertain the notions of good and bad. "Every art and every enquiry, and similarly every action and pursuit, is thought to aim at some good; and for this reason the good has rightly been declared to be that at which all things aim" (11). "The problem of system improvement (as opposed to subsystem improvement) is the problem of the 'ethics of the whole system'. In some sense this use of the term 'ethics' may seem unusual, because ethics is a term often used to connote concepts of good and bad with respect to individual conduct. Indeed, in ordinary discourse the basic underlying notion behind discussions of ethics is closely related to blame and praise ... Now, of course, the discussion of individual behavior does properly belong under the theme described here as the ethics of large-scale systems ... We could therefore sensibly ask whether an individual might have lived his life in a better way than he did, given the resources made available to him by the whole system. The point, however, is that we cannot judge improvement in an individual unless we have an understanding of the nature of the whole system in which the individual lives" (12).

Unfortunately, there is little in our educational system to equip us to handle ethical problems which arise when one takes the results of well-defined problems and attempts to examine them within the ethos from which the problem was extracted. Certainly, there are no algorithms available which would allow us to automate this class of decision-making. Western civilizations appear to have devolved ethical decision-making onto politicians who seem to be rather inadequately equipped to be the specialist in this area -- something which has bothered man for many centuries: "I want to know what is characteristic of piety which makes all actions pious ... that I may have it to turn to, and to use as a standard whereby to judge your actions and those of other men" (13).

Where does this leave systems analysis and computer-aided design? Should they be pushed aside because they are unable to perform in certain ways, and if so, what is to replace them? The advantage of systems analysis is that it takes care of the 'arithmetic' of a particular phase in problem solving, namely, that of examining the effects of changing values of variables within a bounded area and doing so in ways in which the human mind cannot compete. What comes out of this is not 'the solution' but something quite different: the results of this process build up the experience of the system user and hence allow him to learn at considerably less expense than trying to do the same in reality, rather than with some model. It has been argued that what can be handled this way are not the difficult problems (5), but this does not compromise the integrity of the approach. Rather what is needed is some procedure whereby the well-defined problem and its manipulations can be put back into the original value system and hence its original context so that the ethical decisions can be made. So far, only man seems to be able to make these decisions, hence, the need arises to include man in a computer-aided design system.

Computer technology has provided means whereby this may be partially achieved through the use of interactive computing. The use of interaction alleviates one of the earlier objections for it allows the computer-aided design system user, whether he is a professional or lay 'designer' to include his own subjective value system as part of the evaluation (14). Obviously, the level of interaction would need to be much greater than currently offered through the use of existing interactive languages (15).

The difficulty of deciding how to define the boundary of problems and then to allow exogenously defined parameters to become endogenous, particularly when they have only been implicitly defined is not assuaged through the use of interaction. A number of avenues have been investigated with varying degrees of success (16); dialectics seems to be one fruitful possibility (17) although it works better in Marxist societies. The difficulty is in explicating the Weltanschauung of the model builder - we are all limited by our imaginations and are bound by our experiences and what we have learned - "the fact that our knowledge can only be finite, while our ignorance must necessarily be infinite" (18).

Ethics provides the boundary for any model, the challenge, therefore, is to develop procedures whereby models can be built to that boundary. Every model built within that boundary tells something about the set contained within that model but nothing about things beyond it. Popper, in talking about knowledge generally raises the ethos problem: "The traditional systems of epistomology may be said to result from yes-answers and no-answers to questions about the sources of our knowledge. They never challenge these questions, or dispute their legitimacy". The value of computer-aided design systems lies in their abilities to manipulate well-defined problems without having to worry about the 'arithmetic'; with the results of such manipulations one can test hypotheses and learn more about ourselves and our world without necessarily being able to solve all ethical problems in this manner.

ACKNOWLEDGEMENTS

The author wishes to thank Dr. Charles Babendreier, of the National Science Foundation, who encouraged the presentation of this paper, and the National Science Foundation and the Harkness Fellowships which made its presentation possible.

REFERENCES

1. CHURCHMAN, C.W.: Challenge to Reason. McGraw-Hill, New York, (1968), p.2.

2. RUSSELL, B.: The Elements of Ethics - The Subject Matter of Ethics. New Quarterly, February, (1910).

3. WEINWURM, E.H.: Limitations of the Scientific Method in Management Science, Man. Sci., 3, 3, (1957), p. 225

4. CHURCHMAN, C.W.: Letter to the Editor, Man. Sci. 14, 4 (1967), p. 141.

5. RITTEL, H. and WEBBER, M.: Dilemmas in a General Theory of Planning, Policy Sciences, 4, (1973), p. 155.

6. WEINWURM, E.H.: op. cit.

7. ARROW, K.J.: Social Choice and Individual Values. Wiley, New York, (1963).

8. SHELLY, M. and BRYAN, G. (eds): Human Judgments and Optimality. Wiley, New York, (1964).

9. VLIETSTRA, J. and WIELINGA, R.F. (eds): Computer-Aided Design. North Holland, Amsterdam (1973).

10. MASLOW, A.H.: The Psychology of Sciences: A Renaissance. Gateway, Chicago, (1970), p. x.

11. ARISTOTLE: Nicomachean Ethics; Book I, Chapter 1.

12. CHURCHMAN, C.W.: Ref. 1, pp. 4-5.

13. PLATO: Euthypro: Book VII (Socrates).

14. GERO, J.S.: A System for Computer-Aided Design in Architecture. Computer-Aided Design. J. Vlietstra and R. Wielinga (eds). North-Holland, Amsterdam, (1973), p. 307.

15. GERO, J.S., JULIAN, W. and HOLMES, W.N.: Interaction, Interfaces and Design, Eleventh Design Automation Workshop, ACM, N.Y. (1974).

16. BETZ, F.: On the Management of Inquiry. Man. Sci., 18, 4, (1971), p. B117.

17. MASON, R.O.: A Dialectical Approach to Strategic Planning, Man. Sci., 15, 8, (1969), p. B403.

18. POPPER, K.: Conjectures and Refutations: The Growth of Scientific Knowledge. Harper Torchbooks, N.Y., (1965), p. 28.

SURVEY OF DESIGN AUTOMATION IN UNIVERSITIES.

by W.M. vanCleemput
Dept of Computer Science
University of Waterloo
Waterloo, Ontario, Canada.

INTRODUCTION.

Since Design Automation is beginning to emerge as a discipline within the university, SIGDA (the Special Interest Group on Design Automation of the Association for Computing Machinery) decided to conduct a survey of Design Automation activities in Universities.

A questionnaire was published in the SIGDA Newsletter (vol 4, no 3) in September 1974.
In addition, a mailing list of about 200 people working on Design Automation in a university environment was collected.

The response to the survey was gratifying. A total of 39 questionnaires was returned, covering the activities of almost 100 researchers in 32 different institutions.

SUMMARY OF RESULTS.

Question 1.

- Breakdown of responses according to country of origin:

U.S.A. : 20	Canada : 3
U.K. : 9	France : 2
New Zealand: 1	West Germany: 2
Israel : 1	

- Membership in ACM, IEEE or SIGDA:

ACM : 17
IEEE : 16
SIGDA: 8

Appendix 2 contains an alphabetical listing of all respondents.

Question 2.

Areas of design automation:

for each area, a listing of universities and people involved in this area is given in Appendix 1.

Since only information obtained from the questionnaires was used, this listing is rather incomplete.
As could be expected, the most active area is that of digital systems design (A1 - A5), in which 17 groups (out of 32) are active.

Question 2.2

Courses taught: at 10 institutions, introductory courses (undergraduate level) are taught, while at 18 more advanced courses (senior under-graduate or graduate) are available.

Question 2.3

Hardware available:

All groups have access to a large scale, central computing facility. Nevertheless, 23 groups reported having at least one dedicated minicomputer at their disposition.
Only 13 reported having an interactive graphics terminal while respectively 8 and 3 groups used a plotter and digitizer.

Question 2.4

A large number of groups reported having software packages, developed by them, that could be made available to others.
Publishing this information at this time might result in rather incomplete descriptions.
Therefore, it is hoped that SIGDA will be able to create a mechanism, whereby the availability of these packages will be publicized in the SIGDA Newsletter.

CONCLUSIONS.

Although this questionnaire had a rather limited distribution, the response to it was very gratifying.

The general consensus was that Design Automation in Universities should concentrate on the more academic aspects, such as developing algorithms and formal tools. Universities should not try to develop large real-life Design Automation systems unless some valuable experience can be gained by doing so. The construction of production-type DA systems should be left to the industry.

Many of the respondents reported that they had contacts with the industrial environment and thought this to be extremely useful and in most cases essential.

Students in the various engineering disciplines should be taught how and when to use design automation tools within the framework of a total design methodology.

What role should SIGDA play ? The major feeling was that SIGDA should provide a forum where educators can keep informed. It is felt that SIGDA should try to play a role in making it easy to obtain e.g. software packages from other universities. One way of doing this is to publish descriptions of such packages in the Newsletter.
Another useful role for SIGDA is to inform the community about projects being undertaken at the various institutions. Several respondents have agreed to contribute a short description of their work to the Newsletter. It is hoped that this will stimulate interaction between the various DA groups.

I gratefully aknowledge the valuable cooperation of the respondents.

Appendix 1.

The following provides, for each area of Design Automation, a listing of Universities together with the people involved in each area. Only information obtained from the questionnaire was used. This has resulted in a rather incomplete list.

A. ELECTRICAL ENGINEERING.

A.1. Fault Diagnosis and Fault Test Generation.

U.S.A.

Carnegie Mellon University : J. Grason, D. Siewiorek
University of South Florida : O.N. Garcia
University of Southern California : M. Breuer, A.D. Friedman, J.P. Hayes
University of Wisconsin, Madison : C.R. Kime
Southern Methodist University : J.L. Fike
University of Arizona : F.J. Hill

England

University of Southampton : R.G. Bennetts
University of Bath : S.L. Hurst

France

CNRS - Toulouse : P. Azema, M. Diaz

A.2. Logic Design Algorithms (incl. design languages)

U.S.A.

Carnegie Mellon University : J. Grason, D. Siewiorek, M. Barbacci
University of Southern California : M.A. Breuer, A.D. Friedman, J.P. Hayes
University of Wisconsin, Madison : D.L. Dietmeyer
University of Maryland : Y. Chu
Southern Methodist University : J.L. Fike
Case Western University : H.W. Mergler
University of Arizona : F.J. Hill

Canada

University of Waterloo : J.G. Linders
University of Calgary : D. Zissos

England

Kingston Polytechnic : A.A. Kaposi, B.J.L. Saxby, L. Popovic
University of Bath : S.L. Hurst

West Germany

Technische Universitaet Berlin : R.Hoffman

France

CNRS - Toulouse : P. Azema, M. Diaz

A.3. IC and PCB Layout

U.S.A.

Columbia University : O. Wing
University of Southern California : M.A. Breuer, A.D. Friedman, J.P. Hayes
University of Wisconsin, Madison : H. Guchel, D.L. Dietmeyer
Naval Postgraduate School : U.R. Kodres
Southern Methodist University : J.L. Fike
Case Western Reserve University : H.W. Mergler

Canada

University of Waterloo : J.G. Linders, J.A. Smith, W.M. vanCleemput

England

University of Southampton : K.G. Nichols

West Germany

Universitaet Karlsruhe : D.A. Mlynski

A.4. Software Design Automation

U.S.A.

Carnegie Mellon University : W. Wulf
University of Maryland : Y. Chu
Naval Postgraduate School : U.R. Kodres

A.5. Microprogramming Design Automation

U.S.A.

Carnegie Mellon University : D. Siewiorek, M. Barbacci
University of South Florida : O.N. Garcia

West Germany

Technische Universitaet Berlin : R. Hoffman

France
CNRS - Toulouse : M. Diaz

A.6. Computer - aided circuit analysis

U.S.A.

Carnegie Mellon University : R. Rohrer
Columbia University : O. Wing
University of South Florida : O.N. Garcia
University of Houston : G.F. Paskusz
Southern Methodist University : Dr Peskari

Canada

University of Waterloo : J. Vlach, P. Bryant, P. Roe, F. Branin
McMaster University : J. W. Bandler

England

University of Southampton : K.G. Nichols
Kingston Polytechnic : G. Rzevski, D.H. Kail
University of Bath : S.L. Hurst

West Germany

Universitaet Karlsruhe : Prof. D.A. Mlynski, Dr Reiss

A.7. Other

U.S.A.

University of Missouri, Columbia : C. Harlow : computer aided testing

England

Kingston Polytechnic : A.Kaposi, H. Bali, C.D. Partridge, G.W.Shaw :
Reliability engineering, mathematical modelling

B. MECHANICAL ENGINEERING.

U.S.A.

Carnegie Mellon University : S. Paul

Canada

University of Waterloo : D. French, G. Andrews

England

Cambridge University : I.C. Braid, A.R. Grayer
University of East Anglia : A.R. Forrest
University of Strathclyde : T. Allan

C. CIVIL ENGINEERING

U.S.A.

Carnegie Mellon University : S. Fenves
California Polytechnic State University : J.G. Pohl

New Zealand

University of Canterbury : D.G. Elms, P.J. Moss

D. ARCHITECTURE

U.S.A.

Carnegie Mellon University : C. Eastman, D. Stoker, A. Baer, J. Lividini
California Polytechnic State University : J.G. Pohl, D. Grant, N. Greene, P. Winninghoff
Syracuse University : S.A. Coons, J.L. Posdamer
Harvard University : E. Teicholz

Canada

University of Waterloo : T. Bjornstad

England

University of Reading : J. Paterson
University of Edinburgh : A. Bijl
University of Strathclyde : T.W. Maver

New Zealand

University of Canterbury : D.G. Elms

E. CHEMICAL ENGINEERING

U.S.A.

Carnegie Mellon University : G.J. Powers

F. COMPUTER SCIENCE ASPECTS

F.1. Computer Graphics for Design Automation

U.S.A.
Carnegie Mellon University : R. Reddy, C. Eastman
University of North Carolina, Chapel Hill : J.D. Foley
Syracuse University : S.A. Coons, J.L. Posdamer
University of Utah : R.F. Riesenfeld
California Polytechnic State University : J.G. Pohl, J. Beug, P. Winninghoff, N. Greene, D. Grant

England

Cambridge University : R.C. Hillyard, H. Stewart, D.J. Garnett, P. Cross
University of East Anglia : A.R. Forrest, Milne
University of Reading : J. Paterson
University of Edinburgh : M.S. Spring
University of Strathclyde : T. Allan, T.W. Maver

West Germany

Technische Universitaet Berlin : Prof. Lemke

F.2. D A Data Bases

U.S.A.

Carnegie Mellon University : A. Newell, J. Buchanan, C. Eastman, J. Lividini,
D. Stoker

England

Cambridge University : I.C. Braid
University of East Anglia : Stocker, Dearnley
Kingston Polytechnic : K.S.H. Halstead
University of Reading : J. Paterson

Appendix 2.

List of respondents

T. ALLAN, Dept of Design and Drawing, University of Strathclyde, Glasgow, U.K.
P. AZEMA, Dept of Electronic Engineering and Computer Science, LAAS-CNRS, Toulouse, France
J.W. BANDLER, Dept of Electrical Engineering, McMaster University, Hamilton, Ontario, Canada
A. BIJL, Dept of Architecture, University of Edinburgh, Edinburgh, U.K.
T.E. BJORNSTAD, Dept of Architecture, University of Waterloo, Waterloo, Ontario, Canada
I.C. BRAID, Dept of Computer Laboratory, University of Cambridge, Cambridge,U.K.
M. BREUER, Dept of Electrical Engineering, University of Southern California, Los Angeles, California 90007, U.S.A.
Y. CHU, Dept of Computer Science, University of Maryland, College Park, Maryland 20742, U.S.A.
M. DIAZ, Dept of Computer Science, LAAS-CNRS, Toulouse, France
D.L. DIETMEYER, Dept of Electrical and Computer Engineering, University of Wisconsin - Madison, Madison, Wisconsin 53713, U.S.A.
C.M. EASTMAN, Dept of Architecture, Computer Science and School of Urban and Public Affairs, Carnegie Mellon University, Pittsburgh, PA 15213, U.S.A.
D.G. ELMS, Dept of civil engineering, University of Canterbury, Christchurh, New Zealand
J.L. FIKE, Dept of Electrical Engieering and Compuer Science, Southern Methodist University, Dallas, Texas 75275, U.S.A.
J.D. FOLEY, Dept of Computer Science, University of North Carolina at Chapel Hill, Chapel Hill, N.C. 27514, U.S.A.
A.R. FORREST, School of Computing Studies, University of East Anglia, Norwich, U.K.
O.N. GARCIA, Dept of Electrical and Electronics Systems, University of South Florida, Tampa, Florida 33620, U.S.A.
J. GRASON, Dept of Electrical Engineering, Carnegie Mellon University, Pittsburgh, PA 15213, U.S.A.
C. HARLOW, Dept of Electrical Engineering, University of Missouri, Columbia, Mo 65201, U.S.A.
R. HOFFMAN, Informatik - Forshungsgruppe, Technische Universitaet Berlin, 1 Berlin 10, West Germany
S.L. HURST, School of Electrical Engineering, University of Bath, Bath, U.K.
A.A. KAPOSI, School of Electrical and Electronic Engineering, Kingston Polytechnic, Kingston, Surrey, U.K.
C.R. KIME, Dept of Electrical and Computer Engineering, University of Wisconsin, Madison, Wisconsin 53706, U.S.A.
U.R. KODRES, Computer Science Group, Naval Postgraduate School, Monterey, California 93940, U.S.A.
T.W. MAVER, Dept of Architecture and Building Science, University of Strathclyde, Glasgow, U.K.
H.W. MERGLER, Dept of Electrical Engineering, Case Western Reserve University, Cleveland, Ohio 44106, U.S.A.
D.A. MLYNSKI, Dept of Electrotechnic, Universitaet Karlsruhe, D75 Karlsruhe, West Germany
J.B. MUSKAT, Faculty of Mathmatics, Bar-Ilan University, Ramat-Gan, Israel
R.G. NICHOLS, Dept of Electronics, University of Southampton, Southampton, U.K.
G.F. PASKUSZ, Cullen College of Engineering, University of HOuston, Houston, Texas 77004, U.S.A.
J. PATERSON, Dept of Construction Management, University of Reading, Reading, U.K.
J.G. POHL, School of Architecture, California Polytechnic State University, San Luis Obispo, Ca 93407, U.S.A.
J.L. POSDAMER, Dept of Systems and Information Science, Syracuse University, Syracuse,N.Y. 31210, U.S.A.

G.J. POWERS, Dept of Chemical Engineering, Carnegie Mellon University, Pittsburgh, Pa 15213, U.S.A.

R.F. RIESENFELD, Dept of Computer Science, University of Utah, Salt Lake City, Utah 84112, U.S.A.

D.P. SIEWIOREK, Dept of Computer Science , Carnegie Mellon University, Pittsburgh, Pa 15213, U.S.A.

E. TEICHOLZ, Laboratory for Computer Graphics, Harvard University,Cambridge, Mass 02138, U.S.A.

W.M. VANCLEEMPUT, Dept of Computer Science, University of Waterloo, Waterloo, Ontario, Canada

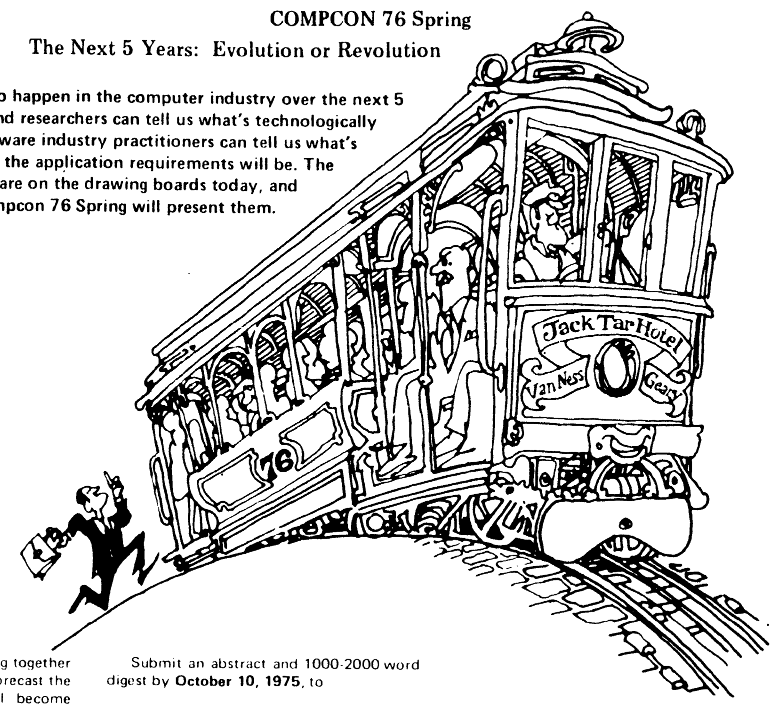
O. WING, Dept of Electrical Engineering and Computer Science, Columbia University, New York, N.Y. 10027, U.S.A.

call for papers

COMPCON 76 Spring

The Next 5 Years: Evolution or Revolution

What's going to happen in the computer industry over the next 5 years? Academicians and researchers can tell us what's technologically possible; hardware and software industry practitioners can tell us what's feasible; users can tell us what the application requirements will be. The real answers to the question are on the drawing boards today, and Compcon 76 Spring will present them.



COMPCON 76 Spring will bring together researchers and practitioners to forecast the products and services that will become practical reality over the next 5 years. Papers are sought on such topics as the following:

- Is "distributed computer systems" just a buzzword?
- Does the supercomputer have a future?
- What will the microprocessor do for us?
- Who will solve our social problems?
- Where is software/firmware/hardware technology going?
- Is computer science related to computing?
- Can computers solve the energy crisis?

In accordance with standing COMPCON policy, exceptional papers not directly related to the conference theme are also solicited.

Submit an abstract and 1000-2000 word digest by **October 10, 1975**, to

Dr. Herschel Loomis,
COMPCON 76 Spring Program Chairman
Department of Electrical Engineering
University of California
Davis, CA 95616

Best paper competition The five best presentations at COMPCON 76 Spring will be awarded special prizes $150 1st prize, $100 2nd prize, $50 3rd prize, plus certificates of award for 4th and 5th prizes--and will be published in **full paper form** in a special issue of *COMPUTER* during 1976. Judging will be based on paper content and quality of presentation.

Short notes Recent work or experience may be reported (without publication) in the "Short Notes" session, a traditional Compcon feature. Short notes are restricted to 5-minute presentations. Informal discussion is encouraged following each presentation. No sales pitches. Submit a 200-300 word summary together with your name and affiliation by **January 1, 1976**, to

Prof. Martin Graham
Dept. of Electrical Engineering
and Computer Science
University of California
Berkeley, CA 94720

FEBRUARY 24-26 **spring**

compcon 76

12th IEEE COMPUTER SOCIETY CONFERENCE
JACK TAR HOTEL, SAN FRANCISCO, CALIFORNIA

IEEE COMPUTER SOCIETY CELEBRATES 25TH ANNIVERSITY

DESIGN AUTOMATION IN ARCHITECTURAL AND BUILDING DESIGN

Review of Some Recent Relevant Books

THE ARCHITECT AND THE COMPUTER
by Boyd Auger. Pall Mall Press, London, 1972, ill, index, 135 pp.

There are still very few books on computers in architecture although there are a number in preparation. Of those already published only two, "Computers in Architectural Design" by David Campion and "Computer Applications in Architecture and Engineering" by G. Neil Harper merit attention. Campion's book begins with the hardware and software and proceeds to applications, with considerable amount of space devoted to the former. Harper's book, as its title indicates, spends most of its time on applications. Boyd Auger has written a book which lies somewhere in between these two. The gamut of the book can be ascertained from a list of the chapter headings: The Architect; The Computer; The Design Office Computer; BASYS; BASYS Applications; Computers in Design; and Prospects and Conclusions.

This is a book for the neophyte who wants a broad, if somewhat shallow, introduction. It differs from both Campion's and Harper's books in that primarily one system combining both hardware and software is described. In a volume as thin as this, it is hard to do justice to the many topics introduced and the choice of some of the applications to be described is mysterious. The Gyroton Structure produces an interesting photograph but tells us very little about the architect and the computer.

"The Architect and the Computer" is an easy book to read; written by an architect for other architects, it will be useful for the novice who if he is interested further will have to go elsewhere for more information.

J.S. GERO

COMPUTER-AIDED DESIGN
by J. Vlietstra and R.F. Wielinga (eds.). North-Holland Publishing, Amsterdam, 1973. ill., index, viii + 462 pp.

Computer-Aided Design: what is it; what effect is it having on design; what areas does it cover; what can I expect from it as a designer; is there a difference between it and computer-augmented design; are there special languages involved; and what effect will it have on me? It is to many of these questions that the contents of this book are directed.

During October, 1972, IFIP supported a working conference on computer-aided design (CAD); this volume presents both the papers and the discussion from that conference. The scope of the conference was defined as follows:

a. Definition and analysis of CAD in order to identify those aspects of the design process and computer technology that combine to produce successful CAD applications.

b. Software "philosophy": systems software and applications software; methodology of building CAD systems; CAD-oriented data structures and the relation with more general data bases; transportability, evolutionary properties and "lifetime" of software.

c. Language: the languages in which CAD systems are written; the language used in communication with the system: the interface between the user and the system programmer.

d. Techniques: input/output organization; data structure considerations; the coupling with manufacturing systems.

e. The economics of CAD: how to cost the various techniques; cost-returns ratio evaluated for means and techniques; how to treat intangibles like the value of time saved, better design and product, integration and overall improvement in the design/manufacture/sales system.

f. Hardware configuration options; peripheral equipment; access-time considerations; interactive versus batch computing; graphic and non-graphical output.

g. Analysis of particular CAD application; essential features of wider potential use; user's job satisfaction; the impact of CAD on the working environment and implications for society at large.

The nineteen papers presented cover most of these topics and, unlike many conferences on CAD, an attempt has been made to examine design and/or synthesis problems rather than simply considering design as continued analysis. The papers range from overview descriptions of applications to detailed descriptions of specific applications; from data structuring problems to interactive languages; and from reliability analysis to problem solving methods.

Some of the papers concerned with building design will be familiar to architects, particularly those by Hoskins, Bernholtz and Meager. Some of the other papers pose fundamental questions concerning the applicability of CAD to certain tasks. Perhaps one of the most interesting general papers is by Powers, who puts forward three representations for non-numerical problem solving: state space; decomposition and theorem-proving. State-space representations are only now beginning to be used to supplant decomposition representations in building design, although they have already found application in planning; dynamic programming and control theory provide suitable manipulative techniques.

This is not a textbook but the proceedings of a conference, hence the presentation is somewhat uneven both in terms of content and appearance. However, architects, designers and planners who are interested in finding out the state-of-the-art would do well to read this book for both the papers and the subsequent discussions are equally enlightening.

J.S. GERO

COMPUTERS IN ARCHITECTURAL PRACTICE

by B. Guttridge and J.R. Wainwright, Crosby Lockwood Staples, London, 1973, 121 pp.

It is not uncommon to find books which lack either literacy or technical rigour. Thankfully, books which are neither literate nor technically rigorous are extremely rare. In this little offering, however, we have a real collector's item.

The first paragraph of the book reveals that:

There are two types of computer:

(a) Analogue (b) Digital

From there the text becomes marginally less revelationary and significantly less accurate. Did you know, for instance, that "In order to produce an efficient design acceptable to the client, the architect must have access to sufficient data to make this possible?" Did you further know that "In order to utilise fully the resources of an office, the architect is obliged to seek methods by which his efficiency can be increased?" You didn't? No wonder the built environment's a mess. But wait, here comes the computer to the rescue: ".... It will calculate the heat loss. It could even estimate the area of radiator panels if asked. Clearly this is a very imaginative program for a computer to handle". Clearly; gosh! If all of this overwhelms the reader, he can turn for help to the glossary. "Off-line:- Description of a method of working where units are connected to the central processor of the computer." I wonder, then, what "on-line" means; pity it doesn't appear in the glossary. Or what about "Memory:- Computer store", presumably the shop in which you can buy a computer which will handle all those imaginative-type programs.

The chapters - which one has to acknowledge are numbered in the right order - purport to cover such topics as Design and Job Management (two programs are described); Applications to System Building (which, although it does not deal with applications to system building, does contain the literary highlight of the opus - "It would appear that system building does have good potential for computer aiding".) Case Study I (which is not a case study but mainly comprises a large volume of poorly explained print-out); Case Study II: a Practise Not Using Computer Application (which is about a practice not using computer application); and General Conclusions (which contains no conclusions, general or specific). The opening chapter claims to be a General Synopsis (is there such a thing as a <u>specific</u> synopsis?); this includes a summary of the work of <u>one</u> University Department (incorrectly reproduced from a standard handout) and what would appear to be a lengthy epistle from someone in the U.S.A.

The whole effort would be derisory were it not for the fact that it will do positive harm in the profession. In the preface, the authors state that the computer is a high-speed moron; at the time of writing this review, the computer was not available for its opinion of the authors.

T. MAVER

RECENT PUBLICATIONS.

compiled by

W.M. vanCleemput
Dept. of Computer Science
University of Waterloo
Waterloo, Ontario, Canada.

1. BASIC QUESTIONS OF DESIGN THEORY
Proceedings of the Symposium on Basic Questions of Design Theory, Columbia University, New York, May 30-31, 1974.
Edited by William R. SPILLERS, Columbia University.
North Holland Publishing Co., 1974; 542 pages
Price : US $ 34.75

Contents:

Part 1: Civil Engineering and Architecture.
Architectural design theory : Models of the design process (V. Bazjanac)
The rule of system identification in design (J.G. Beliveau).
Linguistics and semiotic models in architecture (M. Gandelsonas).
From the basics to a conflict analysis in the treatment of transportation assignment: User vs. system planner (W.R. McShane and C. Marchall).
Limits to the embodiment of basic theories (N. Negroponte).
A strategy for design research (P.A. Purcell, G.L. Mallen and P.G. Goumain).
Is automated architectural design possible? (M.G. Salvadori).
Some problems of structural design (W.R. Spillers).

Part 2: Chemical Engineering.
Design in chemical engineering (D.M. Himmelblau).
Recent developments in process design (R.S.H. Mah).
Design theory: A chemical engineering view (R.L. Motard).
A theory for chemical engineering design (G.S. Powers and D.F. Rudd).
The synthesis problem with some thoughts on evolutionary synthesis in the design of engineering systems (A.W. Westerberg).

Part 3: Mechanical Engineering.
Design Aspects of parameter estimation (J.V. Beck).
Fluid film bearing design: state of the art (V. Castelli and J. Pirvics).
Synthetic dynamics: bond graphs in design (D. Karnopp).
Kinematic structure of mechanisms (F. Freudenstein and L.S. Woo).
Calculus of screws (A.T. Yang).

Part 4: Electrical Engineering.
Exploiting parallelism in design algorithms (D.A. Calahan).
Towards automated design of integrated circuits (S.W. Director).
Extended application of the sparse tableau approach - finite elements and least squares (G.D. Hachtel).
Contour theory and diakoptics (H.H. Happ).

Part 5: Mathematics.
The design of programming languages (S.L. Graham).
Reimann surfaces and the general utilities problem (J.L. Gross).
Graphs and designs (F. Harary).
Some linguistic issues in design (M.A. Harrison).
Multidimensional formal languages (A. Rosenfeld).
Function design (J. Paul Roth).
Bioengineering design (A.K.C. Wong).
A linguistic approach to system analysis (L.A. Zadeh).

Part 6: Socio/Urban Design.
How long can we go on this way ? (S.M. Altman).
Urban design (J.F. Brotchie and R. Sharpe).
Using parsonian structural-functionalism for environmental design (G. Cranz).
Ethics in computer aided design: A polemic (J. Gero).
Computer augmented conceptual design (G.E. Nevill,Jr. and R.A. Crowe).
Formalization of analysis and design in the arts (G. Stiny and J. Gips).

2. PROGRAMMING TECHNIQUES IN COMPUTER AIDED DESIGN.
Edited by M.A. SABIN
Sponsored by:The British Computer Society.
Published by: NCC Publications, 1974.

Contents

3. PROCEEDINGS OF THE NTG MEETING "DESIGN AUTOMATION FOR DATA PROCESSING SYSTEMS AND THEIR COMPONENTS"
Karlsruhe, Germany, September 25-27, 1974.

All papers in German.
Published by VDE Verlag, Bismarckstrasse 33, 1 Berlin 12, Germany.
Price: DM 56.

Contents.

Circuit Design.
Survey and introduction (D. Seitzer).
Computer aided circuit design with universal circuit analysis programs (U. Steinkopf).
Adaption of a simulation program for integrated MOS circuits to process and dimensional effects. (H. Sibbert, B. Hofflinger).
A two-dimensional model for the calculation of bipolar transistors (W. Heinze).
Automatic optimization of digital circuits (R.W. Mitterer).
Problems and mathematical methods for the automated design of circuit layouts (H.J. Schneider).
A method for the partitioning of switching circuits under special boundary conditions (H.J. Groger).
AUTOMASK - a system for computer aided structural design of IC's. (K.Koller).

Logic Design.
Survey and Introduction (R. Piloty).
An extension to design languages (F. Kopke).
On the minimization of function bundles (J. Beister, R. Ziegler).
A method for the design of dynamic CMOS circuits (K. Sickert).
Petri-Nets for the modelling of switching systems (R. Grieshaber, G. Ulrich, S. Wendt).

Test Methods.
Fault experiments for switching circuits (W. Gorke).
Testing of digital IC's with computer controlled testers (R. Bittl).
Test generation for fault location in sequential circuits with MSI components (B. Ebel).
Fault test generation for switching circuits using a 9-valued model (P. Muth).
A contribution to fault simulation of digital circuits with short-circuit faults (E. Matthias, G. Merz).

4. COMPUTER AIDED DESIGN.
Volume 7, number 1, January 1975.

Published by IPC Science and Technology Press, 32 High Street, Guildford, Surrey, England GU1 3EW.

Contents.
Housing site layout system (A. Bijl and Shawcross).
A graphical interactive room-planner (S.E. Anderson).
Dynamic analysis and balancing of linkages with interactive computer graphics (M.R. Smith).
Another look at computer-aided circuit analysis (D.W. Ingram).
DOCPRO - a dynamic simulator for chemical processes (R. Hughes, R.J. Richardson, A.R.M. Wajih).
Computer simulation model for the design of airport terminal buildings (L.W.W. Laing).
Aiding the Drawing Office to detail plate heat exchangers A.M. Cocks, A.C. Henton).

5. DIGITAL SYSTEM DESIGN AUTOMATION : languages simulation and data base.

Published by Computer Science Press, 4566 Poe Avenue, Woodland Hills, California 91364 ($ 17.95).

Contents.
1. System Level Simulation (M.H. MacDougall).
2. Register Transfer Languages and their translation (D.L. Diemeyer and J.R. Duley).
3. Register Transfer Language Simulation (C.W. Hemming and S.S. Szygenda).
4. Design Automation Aids To Microprogramming(R.E. Merwin).
5. Data Structures, data base and file management (P. Losleben).

5. COMPUTER AIDED DESIGN.

Volume 7, number 2, April 1975.

Contents.

6. PROCEEDINGS OF THE WORKSHOP ON DATA BASES FOR INTERACTIVE DESIGN.

Waterloo, Ontario, Canada September 15-16, 1975.
Sponsored by ACM SIGDA, SIGMOD, SIGGRAPH & Univ. of Waterloo.
Available from: ACM Order Department, P.O. Box 12105, Church Street Station, New York, N.Y. 10249.

WORKSHOP ON DATA BASES FOR INTERACTIVE DESIGN

Waterloo, Ontario, Canada
16-17 September 1975

J.G. Linders

A very successful workshop to discuss the development of data bases for interactive design was held in Waterloo under the auspices of three sigs, namely SIGMOD, SIGGRAPH, SIGDA, as well as the Department of Computer Science from the University of Waterloo. The workshop was attended by over 100 delegates from universities, government, industry and research organizations from both the U.S.A. and Canada as well as France, India, Italy, Japan and Sweden.

The idea of the workshop originated from Chuck Radke, past chairman of SIGDA. In planning discussions with the organizing commitee, it was the intent to develop a forum of specialists from the areas of computer graphics, design automation and data bases to discuss the problems associated with data base developments within the design automation environment. The topics ranged from basic data base notions through to data base developments currently in progress. From the papers presented and the ensuing discussions the workshop was well received.

The format of the workshop included four invited speakers to provide tutorial and state-of-the-art lectures in various related topics. In particular, Chuck Rosenthal gave a survey paper of developments in graphics system while Bill McGee provided a tutorial in data base developments principally from the data structure point of view. Robin Williams discussed recent graphic developments within the IBM labs in San Jose. An outstanding survey of data base development was given by Jim Fry.

The papers delivered were divided into the following sessions: Data Bases for Computer Aided Design, Architecture and Management; Data Base Management Systems for Engineering Design and Automation; Drafting and Automated Design. There was also an open session for short talks by attendees. A number of informal sessions were held in the evening on topics of interest generated from the workshop papers. Many users commented on the timeliness of the workshop and expressed an interest that the workshop theme be repeated again in the near future, especially as developments progress in this area.

The workshop proceedings were published and are available at a cost of $10 per copy from either:

Professor J.A. Smith
Dept. of Computer Science
University of Waterloo
Waterloo, Ontario N2L 3G1,
CANADA (until 30 December 1975)

OR

Association for Computing Machinery
Order Department
PO Box 12105
Church Street Station
New York, N.Y. 10249,
U.S.A.

AN INTRODUCTION TO
CHDL (COMPUTER HARDWARE DESCRIPTION LANGUAGES)

The 1975 International Symposium on CHDL and Their Applications, co-sponsored by SIGARCH, SIGDA, IEEE Computer Society, City University of New York, and in co-operation with Utah State University has been held in New York City during September 3-5, 1975.

Dr. K.E. Iverson (IBM Fellow) was the keynote speaker, and Dr. H. Fleisher (IBM Fellow and IEEE Fellow) spoke at the banquet on "Science and Technology in the Academic and Industrial Worlds."

Computer hardware description languages (CHDL's) can be defined as languages for describing, documenting, simulating, and synthesizing digital systems with the aid of a computer. The main purposes of this symposium are to bring together experts in CHDL's and people who are interested in using CHDL's in various applications such as analysis, synthesis, and documentation of digital systems at the system and/or logic level(s).

Computers have been found to be a very effective tool for aiding the design of digital systems. Therefore, the area of design sutomation has received a great deal of attention. The task of designing a digital system can be considered as consisting of the following steps:

(1) The generation of a system diagram from the specifications of the system to be designed.
(2) The production of detailed logic diagram for each subsystem.
(3) The partitioning of the logic diagram into several units.
(4) The assignment of integrated circuits chips for implementing each unit.
(5) The placing of chips on logic cards and of cards on boards.
(6) The interconnecting of the chips.
(7) The testing of the integrated circuit boards.

Computers have been widely used for aiding steps 4 to 7. A total design automation system requires that steps 1 to 3 be automated. CHDL's can be used for aiding system and logic design as well as partitioning a digital system. A designer can use a CHDL to express his design and leave the tedious uninteresting computations to a computer. Recently there has been an increasing interest in CHDL's because they can bridge the important gaps in design automation.

The process of automated logic design may consist of the following steps:

(1) A designer expresses his design in a CHDL by writing a program consisting of the language statements.
(2) A hardware compiler (translator) checks the syntax of the language statements and reports the errors to the designer for correction. After the errors are corrected, the translator produces a data base to be used by the system simulator and the logic synthesizer.
(3) The system simulator verifies the design at the system level. This will save the large amount of computing time used for simulating everything at the detail gate level. If the system performance is unsatisfactory, the design language statements are modified. If the performance is satisfactory, the next step is taken.
(4) The logic synthesizer (a program) uses the data base produced by the translator, accepts the types and constraints of logic components, and produces a logic diagram.

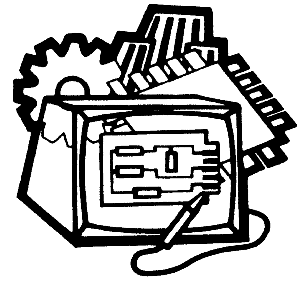
Papers presented in this symposium covered the areas of new CHDL's, compilation and simulation driven by CHDL's, CHDL's in design automations, applications of CHDL's, as well as structured description of digital systems. In addition, there were tutorial sessions dealing with analysis and synthesis of a system. One panel discussion dealt with the future of CHDL's. Another covered the use of CHDL's as the input to design automation systems. There were also two sessions for presenting the most recent results and new concepts which are not fully developed.

Copies of the proceedings can be obtained from either ACM, 1133 Ave. of Americas, New York, New York 10036 or IEEE Computer Society, 5885 Naples Plaza, Suite 301, Long Beach, California 90803.

STEPHEN Y. H. SU, Chairman
1975 International Symposium on
CHDL and Their Applications
Electrical Engineering Department
Utah State University
Logan, Utah 84322
(801)-752-4100 Ext. 7806

13TH
DESIGN
AUTOMATION
CONFERENCE

CALL CALL FOR PAPERS CALL



acm SIGDA

IEEE COMPUTER SOCIETY-DATC

13TH DESIGN AUTOMATION CONFERENCE

REQUIREMENTS FOR SUBMITTING PAPERS

If you plan to submit a paper, you should send three copies of the paper (rough drafts are acceptable) to the program chairman no later than December 12, 1975. Please include a title for the paper plus an abstract (less than 25 words)

Accompanying the draft should be the full name, affiliation address, and telephone number of the principal author, with whom all further direct communication will be conducted.

Notification of acceptance will be sent to you during the first week of February, 1976. After notification of acceptance, you will receive detailed instructions on the format to be observed in typing the final copy. To insure the availability of Proceedings at the Conference, your final manuscript will be due April 20, 1976.

Final papers should be no longer than 5000 words, and the presentation should be limited to 20 minutes. Projection equipment for 35mm slides and viewgraph (overhead projector) foils will be available for every talk. Please indicate what, if any, additional audio-visual aids you require.

Rough drafts are to be sent to the Program Chairman:

Program Chairman
S. A. Szygenda
The University of Texas
Electrical Engineering Dept. (ENS515)
Austin, Texas 78712
512-471-7365

General Chairman
Donald J. Humcke
Bell Laboratories
Building 1F-222
Holmdel, New Jersey 07733
201-949-6253

Sponsors

The sponsors of the Design Automation Conference are the ACM (Association for Computing Machinery) Special Interest Group on Design Automation and IEEE (Institute of Electrical and Electronics Engineers) Computer Society Design Automation Technical Committee.

Design Automation

Design Automation implies the use of computers as aids to the design process.

In the broadest sense, the design process includes everything from specifying the characteristics of a product to meet a marketing objective to enumerating the details of how it is to be manufactured and tested.

Thus design automation embraces applications from one end of the design process to the other.

Site of the 13th DAC
Rickey's Hyatt House
4219 El Camino Real
Palo Alto, California
June 27, 28, 29, 1976

TOPICS OF INTEREST

TECHNIQUES	FUNCTIONS
SOFTWARE AIDS VERIFICATION SIMULATION SYNTHESIS ANALYSIS	INTERCONNECTION PARTITIONING INSPECTION PLACEMENT TESTING LAYOUT
APPLICATIONS	**IMPLEMENTATIONS**
MANUFACTURING DIGITAL SYSTEMS ARCHITECTURE AEROSPACE NAVAL LSI PCB	DATA BASE DESIGN INTERACTIVE TOOLS DESIGN LANGUAGES TOTAL SYSTEMS GRAPHICS

CALL FOR PAPERS

SIGDA SESSIONS

FOR

ACM '76

Houston, Texas   October 20-22, 1976

- Design Automation of Software
- Minicomputers in Design Automation
- Valuing Humans in a Design Automation Environment
- Computer aided Design and Verification of Microprograms
- Interactive Graphics in the Design Automation of Structures and Vehicles
- Automatic Aids in the Layout Process
- Papers of General Interest

Submit Now and Avoid the Rush!

Notify Judith Brinsfield of your Intent.

Computer Aided Design

the use of computers in engineering and building design

An international journal published quarterly by:
IPC Science and Technology Press Ltd.
IPC House, 32 High Street, Guildford, Surrey, England GU1 3EW
Telephone: 0483 71661 Telex: Scitechpress Gd. 859556

A few of the papers scheduled for publication in 1975

Housing site layout system
A. Bijl and G. Shawcross,
Edinburgh University, UK

PBDS—process basic design system
A. Kato et al,
Toyo Engineering Corporation, Japan

The intrinsic method for curve definition
J.A. Adams,
US Naval Academy, Maryland, USA

ICON—the interactive creation of NASTRAN data
A.P. Armit and H.U. Lemke,
Graphical Software Ltd, London

Cybernetic model of the design process
J.H.A.E. Amkreutz,
University of Technology, Eindhoven, The Netherlands

Computer graphics for CAD. Techniques of CAD

Interactive analysis of linear microwave circuits
B.G. Marchent,
University of Warwick, UK

Special Issue:
Computer-aided design in electronics

CAD76

2nd international conference and exhibition on computers in engineering and building design
Imperial College, London 23–25 March 1976
Organized by the journal *Computer Aided Design*

A forum for discussion of significant developments and applications in the use of the computer as a design tool.

Scope of conference:
- Building design
- Civil engineering
- Structural engineering
- Design linked to manufacture
- Shape representation
- Mechanical engineering and plant
- Graphics and displays
- Drafting systems
- Implementing and using c.a.d. systems

Key dates:
- 500-word abstracts by the 2nd June 1975
- Notification of rejection or acceptance by 8th August 1975
- Printed and microfiche proceedings distributed at the conference

The exhibition is showing graphics and other equipment used by the CAD worker together with demonstrations of CAD software.

Details of conference and exhibition can be obtained from:
The Conference Secretary, CAD 76, IPC Science and Technology Press Ltd, 32 High Street, Guildford, Surrey GU1 3EW

Second International Conference and Exhibition on Computers in Engineering and Building Design

CAD76

Imperial College London 23/25 March 1976

Techniques · Design linked to manufacture
Building design · Civil engineering & plant design
Mechanical engineering · Structural engineering

Following the success of CAD 74 the journal *Computer Aided Design* is organizing CAD 76, the 2nd International conference and exhibition – computers in engineering and building design. CAD 76 presents a forum for discussion of significant developments and applications in the use of the computer as a design tool. The fields of manufacturing engineering, building design, structural and civil engineering and plant and process design, will be covered.

Publication of Papers
Papers presented at the conference will be published in the form of printed proceedings. One copy, free of charge, will be given to each delegate at the conference. The proceedings in microfiche form will be offered as an alternative to the printed proceedings.

Conference Language
The conference language will be English.

Conference and Exhibition Venue
Imperial College of Science and Technology
Main College Block
Exhibition Road
Kensington, London SW7 2AZ

Underground.
Piccadilly, Circle and District lines to South Kensington or Gloucester Road

Buses.
9, 52 and 73 to the Royal Albert Hall

Close to the West End, Imperial College is well situated for all facilities

Exhibition
An exhibition of computer graphics equipment and other peripherals used by the CAD worker is to be held in conjunction with the conference. Software suppliers will also be present to demonstrate their CAD programs in the various fields of application. For further information on the availability of exhibition space contact: **John Gregory, Exhibition Manager, CAD 76.**

Exhibition Catalogue
Full details of the exhibition will be contained in a catalogue free to all delegates.

Exhibition Sponsor
The Exhibition is organised by the journal *Computer Aided Design* and sponsored by the publication *Computer Weekly*

Registration
Registration is for £78 + £5.20 VAT for the full conference (3 days) or £38 + £2.00 VAT for a single day. Only one author may register free for each paper. Full conference registration includes an exhibition catalogue, lunches and refreshments, cocktail reception and a copy of the proceedings. One-day registration includes, an exhibition catalogue, lunches, refreshments and a copy of the proceedings. Please use the enclosed registration form. More copies are available from the Conference Secretary.

Early Booking Discount
A discount of £5.40 is offered for early 3 day bookings. If your payment is received by March 1st 1976, the conference fee will be £77.80 including VAT.

Conference Organisers: *Gareth W. Jones and Derek R. Smith*
Conference Secretary: *Mabel Stacey* **Exhibition Manager**: *John Gregory*

CAD 76 IPC House, 32 High Street, Guildford, Surrey, England GU1 3EW
Telephone: Guildford (0483) 71661 Telex: Scitechpress Gd. 859556
Company Registered in England. Registered Number 385206

The programme may be subject to alteration.

IEEE COMPUTER SOCIETY

October 24, 1975

FOR IMMEDIATE RELEASE

PROGRAM SET FOR SYMPOSIUM ON COMPUTER ARITHMETIC

Eight technical sessions for the Third Symposium on Computer Arithmetic have been announced by the chairman, T.R.N. Rao of Southern Methodist University.

Two morning and two afternoon sessions are scheduled for each day of the symposium, to be held November 19-20 at Southern Methodist University, Dallas, Texas, under the co-sponsorship of the IEEE Computer Society's Technical Committee on Computer Architecture and SMU's Department of Computer Science and Operations Research.

The four sessions set for the opening day, Wednesday, November 19, and their chairmen are "Mathematical Foundations of Computer Arithmetic," Ulrich Kulisch; "Control and Monitoring of Precision," Nick Metropolis; "Number Systems," William J. Cody; and "Residue Arithmetic and Error Control," Harvey L. Garner.

Sessions scheduled for Thursday, November 20, are "Arithmetic Algorithms and Their Analysis," chaired by James E. Robertson; "Case Studies of Arithmetic Processor Design and Implementation," Algirdas Avizienis; "Ultra-High Speed Arithmetic for Special Purpose Processors," T.C. Chen; and "Parallelism and Array Logic," Shmuel Winograd.

Fees for advance registration, prior to November 10, are $30 for IEEE and Computer Society members, $35 for non-members, and $10 for students.

(over)

INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS

EXECUTIVE SECRETARY: POST OFFICE BOX 639 / SILVER SPRING, MARYLAND 20901 / TELEPHONE (301) 439-7007
PUBLICATIONS OFFICE 5855 NAPLES PLAZA / SUITE 301 / LONG BEACH, CALIFORNIA 90803 / TELEPHONE (213) 438-9951

Late registrations will be an additional $10 for both members and non-members. The fee includes a copy of the proceedings and, except for students, two buffet luncheons and social hour.

Send registration fee and request for accomodations at the Ramada Inn Central to Prof. John Fike, SCA Local Arrangements Chairman, Department of Computer Science and Operations Research, Southern Methodist University, Dallas, TX 75275; telephone, (214) 693-3081.

About the Symposium

Design Automation has been remarkably developed in recent years and comes to play an important role especially in electronic industry in the USA and Japan.

At this time it would be quite beneficial to provide a forum to exchange information and discuss the technical matters in the field of Design Automation. The subjects on the agenda include Layout and Design, Design Automation Systems, Automatic testing, Design Automation in LSI and Aircraft Design Automation.

We believe that this symposium will do much to imporve the technical level in this field and also to promote friendly relations between the USA and Japan.

USA AD HOC COMMITTEE

Dr. Richard E. Merwin
Assistant for Science & Technology
(Data Processing)
BMD Program Office

Mr. J. M. Galey
IBM Corporation
IBM Bldg. 28 - Dept. H 83

Dr. Charles Radke
Chairman, ACM SIG for
Design Automation
IBM(B99/965)

Mr. Steve Pardee
Chairman, IEEE Computer
Society Technical Committee
on Design Automation
Bell Telephone Laboratories

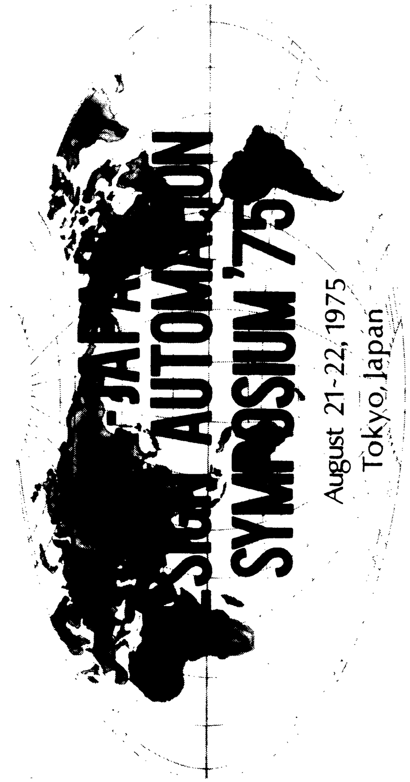
Prof. Steve Szygenda
Chairman, IEEE Computer Society
Technical Committee on Simulation
EE Dept. ENS515
University of Texas

USA-JAPAN DESING AUTOMATION SYMPOSIUM '75 COMMITTEE

General Chairman	Tohru Motooka Dept. of Electrical Engineering University of Tokyo
Vice Chairman	Tadashi Kurachi Design Automation Hardware Development Dvn. Toshiba LTD.
Program Chairman	Junji Tsuda Central Research Lab. Hitachi LTD.
Finance Chairman	Akihiko Yamada Systems Planning Dept. Computer Engineering Dvn. NEC LTD.
Publications Chairman	Takao Hayashi Computer Engineering Dept. Fujitsu LTD.
Arrangements Chairman	Chiyoji Tanaka Computer Works Computer Manufacturing Dept. Mitsubishi LTD.
Correspondence Chairman	Yoshiaki Koga Dept. of Electrical Engineering Defense Academy
Publicity Chairman	Kazuhiro Ueda Integrated Circuit Section Musashino ECL, N.T.T.
IPS Representative	Kazuyuki, Hirakawa Switching and Processing Lab. Engineering Development Dvn. OKI Electric Co. LTD.

Sponsors

USA-JAPAN DA SYMPOSIUM is sponsored by the Information Processing Society of Japan in participation with IEEE Computer Society.



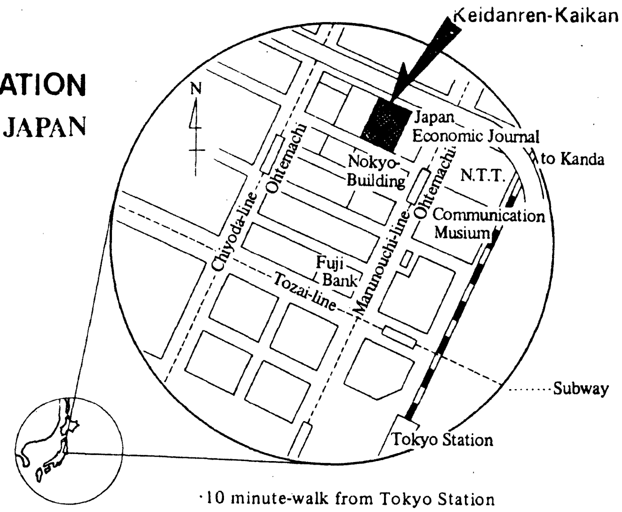
ADVANCE PROGRAM

USA-JAPAN DESIGN AUTOMATION SYMPOSIUM '75 TOKYO-JAPAN

August 21-22, 1975

Keidanren-kaikan

1-9-4, Ohtemachi, Chiyoda-ku
Tokyo, 100 JAPAN
TEL 03 - 279 - 1411



·10 minute-walk from Tokyo Station

Symposium Program

Aug. 20, 1975 (Wed.)	Afternoon	-	Registrations
Aug. 21, 1975 (Thurs.)	Morning	-	Registrations
		-	Opening Sessions
	Afternoon	-	Sessions
		-	Reception
Aug. 22, 1975 (Fri.)	Morning	-	Sessions
	Afternoon	-	Sessions
		-	Closing of Symposium

Proceedings

The proceedings will be made available to Symposium participants in advance of the Symposium.

Additional copies may be purchased at the Symposium.

Sessions

The following sessions will be provided with about 20 timely technical papers.

Session 1 Layout and Design
Session 2 Design Automation Systems
Session 3 Automatic Testing
Session 4 Design Automation in LSI
Session 5 Aircraft Design Automation

People to People program

Languages

English and Japanese can be used in the symposium. Simultaneous interpretation will be made during the symposium.

Advance Registration

Early registration is recommended. Advance registration closes July 15, 1975. After that date, registrations will be accepted at the Symposium registration desk.

Kindly tranfer registration fee to:

(for USA participants)

The Sumitomo Bank Ltd., Fuchu Branch No. 410853
USA-Japan Design Automation Symposium '75 Committee

(for Japanese participants)

住友銀行府中支店 No.410853
日米設計自動化シンポジウム'75委員会

And send the following form to:

(for USA participants)

Mr. Akihiko Yamada
Computer Engineering Devision
N E C Ltd.
1-10 Nitshin-cho, Fuchu-shi
Tokyo, 183 Japan

(for Japanese participants)

〒183 東京都府中市日新町1－10
日本電気株式会社
コンピュータ技術本部
[illegible]

Advance Registration Form
USA-JAPAN DESIGN AUTOMATION SYMPOSIUM '75
Aug. 21-22, 1975 Tokyo, Japan

Name ....
(Last) (First)

Company ....

Mail Address ....
(No.) (Street)

....
(City) (State) (Zip) (Country)

Registration Fee \$50 or ¥15,000

The University of Manitoba
Faculty of Engineering, Department of Electrical Engineering

Extension Division, Department of Professional Studies

NUMERICAL METHODS
OF OPTIMIZATION

With Applications
in Optimal Design

A Five Day Course

DATES: December 8 to 12, 1975

TIME: 9 am to 4:30 pm

LOCATION: Engineering Building
The University of Manitoba

FEE: $300 (includes course materials coffee and opening dinner)

COURSE INSTRUCTOR: Dr. J.W. Bandler, Professor of Electrical Engineering and Coordinator of the Research Group on Simulation, Optimization and Control, McMaster University. Dr. Bandler will be assisted in laboratory sessions by an instructional team from the Department of Electrical Engineering, The University of Manitoba.

This course will provide an opportunity to learn more about the theoretical aspects of modern optimization methods, and to acquire skill in some practical aspects of technique. The programme will include lectures, discussion and a great deal of hands-on experience with prepared user-oriented packages. Each participant will receive a set of notes, reprints, reports, and a choice of one of several user-oriented optimization programmes fully documented with examples and complete programme listings.

A partial list of topics to be covered includes:

Basic concepts concerning objective functions, constraints, minima and maxima, the Jacobian matrix, the Hessian matrix, Lagrange multipliers and nonlinear equations.

Least squares, least pth and minimax schemes. Algorithms involving steepest descent and direct search. Linear searches and conjugate directions.

Formulations and algorithms for nonlinear least squares, least pth and minimax optimization.

Fundamentals of linear programming. Discrete optimization.

Applications to engineering design. Interactive optimization. Computer-aided design of electrical circuits. Analog and digital filter design. Simultaneous design centering, optimal tolerancing and optimal tuning. Worst-case design. Efficient sensitivity evaluation.

Extensive discussion of pitfalls. How to evaluate the usefulness of an optimization programme. How to choose appropriate variables and constraints in a physical problem.

FOR ADDITIONAL INFORMATION AND APPLICATION FORMS CONTACT:

Dr. A. Wexler
Department of Electrical Engineering
(204) 474-9834

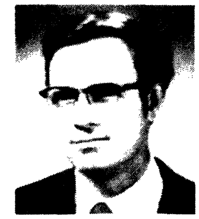
Mr. E.O. Anderson
Extension Division
(204) 474-8207

The University of Manitoba
Winnipeg, Manitoba
R3T 2N2

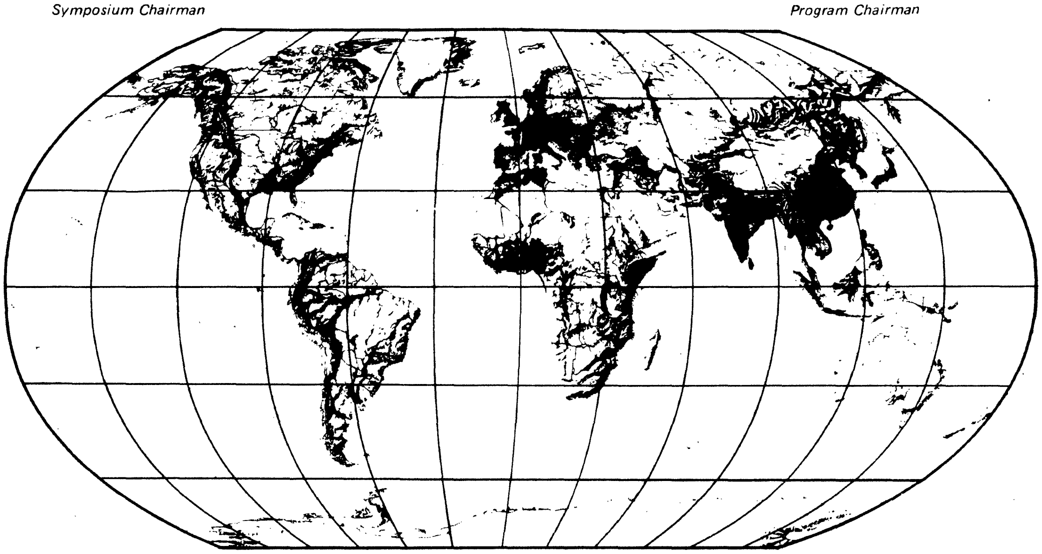
S. Y. H. Su
Symposium Chairman

1975 INTERNATIONAL SYMPOSIUM ON COMPUTER HARDWARE DESCRIPTION LANGUAGES AND THEIR APPLICATIONS

September 3-5, 1975
Graduate Center
City University of New York
33 West 42nd Street
New York City



D. L. Dietmeyer
Program Chairman



Co-sponsored by CUNY, the IEEE Computer Society Technical Committee on Computer Architecture, ACM SIGDA, and ACM SIGARCH, in cooperation with Utah State University, this symposium will bring together experts in CHDL (computer hardware description languages) and people who are interested in using CHDL for describing, documenting, simulating, and synthesizing digital systems with the aid of a computer. The program contains tutorial and research papers with each paper reviewed by at least two referees. There are also two sessions for presenting the most recent results and new concepts which are not fully developed.

Tuesday, September 2

7:00-9:00 p.m. Registration and Get-Acquainted Hour

Wednesday, September 3

8:20-8:30 a.m. Introduction: S. Y. H. Su, Chairman
Welcome: R. E. Marshak, President of CCNY

8:30-9:10 Keynote Address: K. E. Iverson, IBM Fellow

Session I: **STRUCTURED DIGITAL SYSTEMS** (tutorial)
Chairman: R. E. Miller, IBM Watson Research Center, IEEE Fellow

9:10-9:50 "Structured Approach to Digital Systems," S. Patil, Project MAC, MIT

10:10-10:50 "A Directed Graph Model for Hardware/Software Design," F. T. Brashaw, Case Western Reserve University

10:50-11:30 a.m. "Comparison of Graph Models for Parallel Computation and their Extension," S. Y. Fou and G. Musgrave, Brunel University, England

Session II: **DESIGN LANGUAGE** (tutorial)
Chairman: H. Ofek, IBM Watson Research Center

1:00-1:30 p.m. "Updated AHPL (A Hardware Programming Language)," F. J. Hill, U. of Arizona

Session III: **RECENT RESULTS**

1:30-2:30 *Chairman:* J. G. Linders, U. of Waterloo, Canada

Session IV: **WHERE DO WE GO FROM HERE?** (Panel)
Chairman: Y. Chu, U. of Maryland

3:00-4:30 *Panelists:* J. A. N. Lee, R. McClure, F. J. Mowle, R. Piloty, E. Stabler, J. G. Vaucher

4:30-5:30 Planning Session for Future Symposium (Submit proposals to Dr. S. Su, indicating financial support, if any. Proposals should be received by September 1.)

5:30-7:00 Social Hour and Informal Discussions

Thursday, September 4

Session V: **NEW COMPUTER HARDWARE DESCRIPTION LANGUAGES**
Chairman: S. Y. H. Su, Utah State University

8:00-8:30 a.m. "A Model Approach to the Description of Hardware Systems," E. W. Vogel, Generaldirektion PTT, Bern, Switzerland

8:30-9:00 "DIGITEST II: An Integrated Structural and Behavioral Language," F. J. Rammig, Universitat Dortmund, Germany

9:00-9:30 "APL*DS: A Hardware Description Language for Design and Simulation," W. R. Franta and W. K. Gilol, University of Minnesota

Session VI: **CHDLs IN DESIGN AUTOMATION SYSTEMS**
Chairman: J. P. Hayes, USC

10:00-10:30 "The MODEL/LINDA Design Automation System," I. Lewis and A. M. Peskin, Brookhaven National Laboratory

10:30-11:00 "A Hardware Compiler for Interactive Realization of Logical Systems Described in CASSANDRE," Y. Bressy, B. David, Y. Fantino, J. Mermet, Universite Scientifique et Medicale de Grenoble

11:00-11:30 "Applications of an ISP Compiler in a Design Automation Laboratory," M. R. Barbacci and D. Siewiorek, Carnegie-Mellon University

Session VII: **NEW APPLICATIONS OF CHDLs**
Chairman: J. L. Houle, University of Montreal

1:00-1:30 p.m. "The Use of Two CHDL Systems, PMS and DIDL, in the Design of a Fourier Transform Processor," A. M. Despain, Utah State University

1:30-2:00 "A Language for the Specification of Digital Interfacing Problems," A. C. Parker and J. W. Gault, North Carolina State University

2:00-2:30 "Fault Test Generation Using a Design Language," B. M. Huey and F. J. Hill, University of Arizona

Session VIII: **USING CHDL AS AN INPUT TO DESIGN AUTOMATION SYSTEMS** (panel)

3:00-4:30 *Chairman:* J. F. Lund, Honeywell, Phoenix, Arizona
Panelists: M. Barbacci, M. D. Breuer, N. Garaffa, R. L. Hasterlik, D. P. Siewiorek

4:30-6:30 Social Hour and Informal Discussions

7:00-10:00 10-course Chinese Banquet. Speaker: H. Fleisher, IBM Fellow and IEEE Fellow, "Science and Technology in the Academic and Industrial Worlds."

Friday, September 5

Session IX: **EXTENSIONS OF CDL**
Chairman: R. W. Hartenstein, University of Karlsruhe

8:00-8:30 a.m. "A CDL Compiler for Designing and Simulating Digital Systems at the Register Level," J. Bara and R. Born, Michigan Technological University

8:30-9:00 "A Position Paper on Extensions to the Computer Design Language," L. R. Stine and F. J. Mowle, Purdue University

9:00-9:30 "Segmentation Constructs for RTS III, a Computer Hardware Description Language Based on CDL," R. Piloty, Technische Hochschule Darmstadt

Session X: **CHDL DRIVEN SIMULATION**
Chairman: D. L. Dietmeyer, University of Wisconsin–Madison

10:00-10:30 "Simulation of Switching Circuits by SSM—a New Hardware Simulation Language," W. Goerke and H. J. Hoffmann, Universitat Karlsruhe

10:30-11:00 "The HILO Logic Simulation Language," P. L. Flake, G. Musgrave and M. Shorland, Brunel University

11:00-11:30 "LASCAR: A Language for Simulation of Computer Architecture," D. Borrione, ENSIMAG, Grenoble Cedex

Session XI: **STRUCTURE DESCRIPTION**
Chairman: B. J. Smith, University of Colorado

1:00-1:30 "Computer Structure Language (CSL)," J. A. Harris and D. R. Smith, SUNY, Stony Brook

1:30-2:00 "A Structural Modeling Language for Architecture of Computer Systems," R. I. Gardner, Jr., G. Estrin and H. Potash, UCLA

2:00-2:30 "OSM-Microprogrammed Hardware Structure Description Language," R. W. Marczynski, W. T. Pulczyn, J. M. Sochacki, Polish Academy of Sciences, Warsaw

Session XII: **SHORT PAPERS**
3:00-6:00 *Chairman:* G. J. Lipovski, University of Florida

**S. Su will be in Taiwan during July 10-August 31. Information about the Symposium can be obtained from A. A. Sarris, 351 W. 53rd St., New York, NY 10019. Telephone: (212) 743-2404 or 245-6359.*

HOTEL RESERVATION FORM

NAME ____________________

ADDRESS ____________________

☐ $20 single room
☐ $24 double room

Rates do not include local taxes. If a double room is chosen, both names must be given.

Please make reservations two weeks prior to arrival. Rooms cannot be guaranteed at the above rate if reservation is not received by August 20th. Reservations will be held only until 6:00 P.M. unless later hour is specified. Send reservation form to:

Commodore Hotel
Park Ave. at 42nd St.
New York, NY 10017
Attention: Gene Beaver

CHDL SYMPOSIUM REGISTRATION FORM

NAME ____________________
(first) (last)

ADDRESS ____________________

☐ Member of ACM
☐ Member of IEEE or IEEE Computer Society

Number of people attending Spouses' Program ________.

Advanced Registration (Register before August 15th and save $15).

Members (ACM or IEEE):	$55.00*	=	$
Non-members:	$65.00*	=	$
Students:	$10.00**	=	$
No. of Additional Banquet tickets ____	@ $15 each	=	$
	Total		$

**Includes symposium proceedings, 10-course Chinese banquet, coffee.*
***Includes proceedings and coffee.*

Make check payable to: "1975 International Symposium on CHDL"

and mail to: A. A. Sarris
Symposium Treasurer
351 West 53rd St.
New York, NY 10019

1976 acm computer science conference

disneyland hotel / anaheim / california / 10-12 february 1976

short reports	on current research by students, faculty, and researchers in the computer and information sciences/abstracts only are required/deadline for submission of abstracts is 1 december/instructions for preparation of abstracts are in the action packet/
invited talks	robert barton and richard hamming present their views on the nature, potential and limitations of computers and computer science/design options in computer hardware by gordon bell of digital equipment corporation and carnegie-mellon university/speech understanding systems by raj reddy of carnegie-mellon university/
book exhibit	by the leading publishers and distributors of computer science books/
luncheon	for chairs of departments of computers and information sciences/
employment register	provides an opportunity for employers and prospective employees to meet and exchange information/for additional information on the employment register write to orrin e. taulbee/department of computer science/university of pittsburgh/pittsburgh/pennsylvania 15260/
technical symposium	on computer science and education jointly sponsored by the acm special interest groups on computer science education (sigcse) and computer uses in education (sigcue) on 12-13 february 1976 at the disneyland hotel in conjunction with csc 76/for information write to either of the co-chairs/paul lorton jr, university of san francisco, san francisco, california 94117/ron colman, california state university, fullerton, california 92634
registration fee	$30 for the general public/$25 for acm members/$5 for students/advanced registration forms are in the action packet/
action packet	can be obtained by refolding this announcement so the return address is on the outside and mailing it/

NAME ______________________________

ADDRESS ___________________________

____

AFFIX
STAMP

Julian Feldman/Fred Tonge
Department of Information and Computer Science
University of California
Irvine, California 92664

REQUEST FOR ACTION PACKET FOR 1976 ACM COMPUTER SCIENCE CONFERENCE

Computer professional seeking a position?

The

COMPUTER SCIENCE EMPLOYMENT REGISTER

matches computer scientists
and data processing specialists with
employer opportunities.

Employer seeking computer professionals?

The Fourth Annual Register will be held at the Anaheim ACM Computer Science Conference. Further information and official forms may be obtained from:

Orrin E. Taulbee
Department of Computer Science
University of Pittsburgh
Pittsburgh, Pennsylvania 15260

PROCEEDINGS FOR 9th, 10th and 11th DA WORKSHOP

The following is the rate schedule as agreed to by SIGDA and the DA Technical Committee of IEEE.

	COPIES		SIGDA & ACM Members	All Others	Amount
1)	_______	9th DA Workshop Proceedings (1972) @ $10.00 376 Pages	$10.00	$16.00	_______
2)	_______	10th DA Workshop Proceedings (1973) @ $10.00 288 Pages	$10.00	$16.00	_______
3)	_______	11th DA Workshop Proceedings (1974) 379 Pages	$12.00	$20.00	_______
		Member Number ________		TOTAL =	________

Please send your order prepaid to:

ACM Inc.
P.O. Box 12105
Church Street Station
New York, New York 10249

Make your checks payable to Association for Computing Machinery (an added charge is made for billing).

JOIN SIGDA JOIN SIGDA JOIN SIGDA JOIN SIGDA JOIN SIGDA JOIN SIGDA JOIN SIGDA JOIN SIGDA

____
Name (please type or print)

____
Affiliation

____
Mailing Address

____
City State Zip

Annual membership dues are $3.00 for ACM members and $5.00 for others.

____ Enclosed annual dues.

____ Please send more info.

Mail to SIGDA
ACM INC.
P.O. Box 12105
Church Street Station
New York, NY 10249

Association for Computing Machinery
1133 Avenue of the Americas, New York, New York 10036

Non-Profit Org.
U. S. POSTAGE
PAID
New York, N. Y.
Permit No. 2397

```
1050855 00                GDA **
CHARLES E RADKE            JAN20
IBM CORP DEPT 829 BLDG 951
PO BOX 390
POUGHKEEPSIE           NY   12602
```