Not only computing—also art

JOHN LANSDOWN

That toddling town

Once again the ACM SIGGRAPH conference has come and gone (surely, like the Horse of the Year Show seems to, it must occur more frequently than once a year!). Those of you who are members of SIGGRAPH—and anyone with more than a casual interest in computer graphics should be—will have now received their copy of the proceedings; two and a bit volumes of impressive information describing mainly American work completed and in progress. There is no question but that computer graphics is now the current major concern of the US industry: more than 2,500 delegates attended the conference together with over 1,500 representatives of exhibitors, making the SIGGRAPH conference the most popular in the computing field. This year, I was lucky enough to be able to attend. Three reasons prompted my going: firstly, of course, an interest in graphics and animation; secondly, because a personal computer conference was scheduled to be held at the same time; and thirdly, because the conference was to take place in Chicago, a town that I'd always wanted to visit.

For reasons which I was unable to discover, the personal computer conference disappeared without trace: no one seemed to know why or where. But this was perhaps just as well because there was so much to see and hear at SIGGRAPH that it is unlikely that I could have covered both. In addition, more time was left to explore the mythological city known to most of us through gangster films, popular songs, the writings of Carl Sandberg and Upton Sinclair, and the strange conceptions of Brecht (who, incidentally, had never visited the place when he wrote Happy End and St Joan of the Stockyards).

The city holds a special fascination for the architect. It is an outdoor museum of the architecture of the last hundred years displaying the works of the giants Adler, Richardson, Sullivan, Frank Lloyd Wright and Mies van der Rohe. It is the home of the 'Chicago School of Architecture' which, as author Ira Bach tells us in his indispensable *Chicago on Foot,* for the ten years from 1883, pioneered the skyscraper which was then:

'reaching the unprecedented heights of twelve, fourteen, sixteen and 23 stories. The architects of the Chicago school employed a new type of construction: the iron skeleton, at that

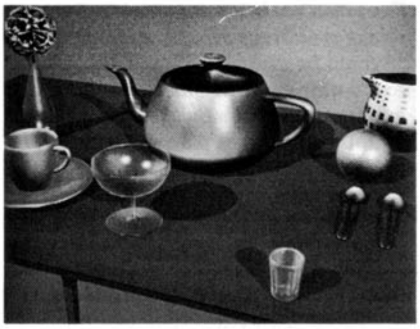
construction". They invented a new kind of foundation to cope with the problems of the muddy ground of Chicago: the floating foundation. They introduced the horizontal elongated window: the "Chicago window". They created the modern business and administration building. And around the turn of the century, the so-called Prairie House came into being here.'

Chicago too, is a living Guiness Book of Records—the world's tallest building (109 floors), fastest lifts, tallest block of flats, largest wholesale market building, first comprehensive city plan, even what is claimed to be the world's busiest corner. Apparently, the ice-cream sundae was invented in a Chicago suburb by a soda-fountain owner who did so, would you believe, to attract business he lost because stringent byelaws banned the sale of ice-cream sodas on a Sunday! But, like Londoners, until recently the inhabitants of Chicago had made little attempt to conserve examples of their past heritage, so that some of the best works of the Chicago school have been demolished. Since the early 'seventies, however, preservation has been taken seriously and, whilst I was there, a magnificent restoration of part of Sullivan's 1899 Carson Pirie Scott building was unveiled.

I was surprised to find that what must be civic pride leads the city to play down its gangster past. I know that the era of Capone, Dillinger and Bugsie Moran was a very short one but it is so firmly set in the public imagination that you think it would be exploited for tourism—but, like the stockyards and the personal computer conference, it seems to have gone into oblivion. But the Wrigley building, the Tribune Tower, the Robie house are all there—see them if you ever get the chance.

More resources than taste

At the conference itself, I sensed the primary preoccupation to be with the problems and potential of raster graphics, particularly to create realistic views and animations. To someone like myself with only limited computing facilites at his disposal (though, nonetheless, having more graphics devices than most), it was disappointing to feel that the main thrust of the current work seems to be in producing completely realistic pictures in full colour with highlights, surface textures and shadows—all regardless of expense



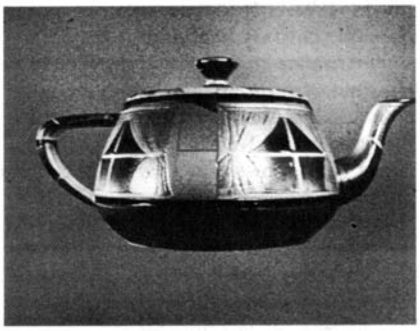


Figure 1 (top)
Figure 2 (above)

things are, I would have preferred more concentration on the development of cheap, fast systems, algorithms and techniques capable of doing economically only a limited subset of the things being shown—but applicable to a wider range of general needs. Few of the workers seem to be caring about the general graphics user who has a mini- or micro-based machine. One of the presentations showed some wonderful graphics, but these were prepared on a system which consisted of three PDP11/34s, one for each colour gun of the display! The computation necessary to send the graphics information to these guns was done by some huge mainframe machine in the background.

From the evidence of some of the presentations, it is no exaggeration to say that, given the money, any image can now be synthesised by computer graphics. Unfortunately, this incredible ability seems to have gone to the heads of many of the workers who, by all appearances, vie with one another to produce the most outlandish and bizarre, not to say tasteless, images. The drawings that many produce, particularly those who claim their work to have an art content, are generally lacking in any sense of design. Self-indulgence reigns. Thankfully, there were one or two items to which, in my view, this criticism could not be applied. In particular, two computer produced

ADDITIONS TO THE BCS LIBRARY JUNE 1979

Advances in computer architecture, G. J. Myers. Wiley, 1978.

The Architecture of concurrent programs, P. Brinch Hansen. Prentice-Hall, 1977.

Architecture and models in data base management systems, G. M. Nijssen. North Holland, 1977.

BASIC programming for scientists and engineers, W. N. Hubin. Prentice-Hall, 1977.

The Chips are down: the future impact of microprocessors and computers on management in Britain, Earth Resources Research, 1978.

Codasyl approach to data base management, T. W. Olle. Wiley, 1978.

Computational mathematics in engineering, S. A. Hovanessian. Lexington Books, 1976.

Computational methods for the solution of engineering problems, G. A. Brebbia and A. J. Ferrante. Pentech Press, 1978.

Computer applications in ship technology, Heyden, 1977.

Computer crime, A. Bequai. Lexington Books, 1978.

The computer industry, J. T. Soma. Lexington Books, 1976.

Computer law, C Tapper. Longman, 1978.

Computer organisation, V. C. Hamacher. McGraw-Hill, 1978.

Computer selection, E. O. Joslin. Technology Press, 1977.

Computer simulation and modelling: an introduction, R. S. Lehman. Lawrence Erlbaum, 1977.

Computer software engineering: proceeding of the symposium, J. Fox, editor. Polytechnic Press, 1976.

Computer storage systems and technology, R. E. Matick, Wiley, 1977.

Data communications—facilities, networks and systems design, D. R. Doll. Wiley, 1978.

Decision Networks, N. A. J. Hastings and J. Mello. Wiley, 1978.

Digital computer electronics, A. P. Malvino. McGraw-Hill, 1977.

Digital continuous system simulation, G. A. Korn and J. V. Wait. Prentice-Hall, 1978.

Genmix—a general computer program for two dimensional parabolic phenomena, D. B. Spalding. Pergamon, 1977.

A guide to PL/M programming for microcomputer applications, D. D. McCracken. Addison Wesley, 1978.

Introduction to problem solving and programming with PASCAL, G. M. Schneider. Wiley, 1978.

Information management systems: database primer, V. C. Prothro. Petrocelli/Charter, 1976.

Job control language, R. Ashley and J. Fernandez. Wiley, 1978.

Long range forecasting: from crystal ball to computer, J. A. Armstrong. Wiley, 1978.

Microprocessors and microcomputer systems, G. V. Rao. Van Nostrand, 1978.

Microprocessors in instruments and control, J. Bibbero. Wiley, 1977.

Modern data processing, R. R. Arnold. Wiley, 1978.

Modern methods for computer security and privacy, L. J. Hoffman. Prentice-Hall, 1977.

PL/C essentials, L .J. Mazlack. McGraw-Hill, 1977.

Principles of compiler design, A. V. Aho and J. D. Ullmann. Addison-Wesley, 1978.

Programming in PASCAL, P. Grogono. Addison-Wesley, 1978.

Real-time computer systems, A. L. Freedman and R. A. Lees. Edward Arnold, 1977.

Risk management in action, J. A. T. Pritchard. NCC, 1978.

RISS: a relational data base management system for minicomputers, M. J. Meldman. Van Nostrand, 1978.

RTL/2: design and philosophy, J. P. G. Barnes. Heyden, 1976.

Systems for large data bases: proceedings of the 2nd International Conference on Very Large Data Bases, P. C. Lockermann and E. J. Neuhold, editors. North Holland, 1977.

Thinking with the teachable machine, J. H. Andreae. Academic Press, 1977.

NOT ONLY COMPUTING—ALSO ART *continued*

films were especially attractive: one, clearly created by a designer used to presenting complex information in an attractive way, was the trailer to the new Walt Disney film, *Black Hole.* This was an extraordinarily simple but effective use of animated line graphics diagrammatically illustrating the concept of a black hole. I would have liked to have shown you a still from this film, but Walt Disney Productions (who didn't make the trailer themselves) couldn't supply me with a copy.

The other was a 25-minute animated film by Nelson Max of the Lawrence Livermore Laboratory called *Space filling curves.* This was one of the few films which arguably could only have been created by computer. Essentially it dealt with the snowflake and Peano curves giving striking visual proofs of their properties—in the one case, the infinite length and the other, the fact that it passes through every point in the square. It is difficult to describe in words how these proofs were achieved except to say that the notional camera used to film them zoomed in at the same speed as the curves were drawn. The effect of this was to achieve a static picture so that, instead of more detail being seen as the camera got closer, the same image occurred again and again. No more potent visual explanation of recursion could be imagined.

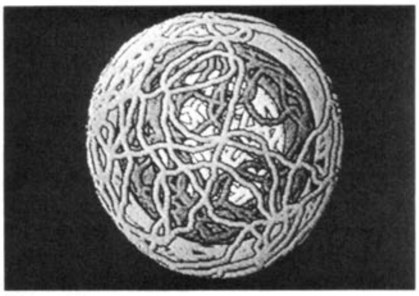


Figure 3

Ken Knowlton

Every year (in theory at least, for nothing we do is very well ordered) the Computer Arts Society chooses an Honorary Life Member on the basis of that person's contribution to computer art. Last year we chose the British artist living in California, Harold Cohen: others are John Whitney, Edward Ihnatowicz, Iannis Xenakis, Knut Wiggens, and Peter Zinovieff. This year our choice was Ken Knowlton, who works for Bell Labs. Surprisingly, though Ken is internationally known, he had never been to Europe until September 1979, when we took the opportunity of having him come and speak to us on his latest work. He showed us a number of films including one on the software keyboard mentioned in these columns before. One of his most interesting films is that called *Baobab* made with Emmanuel Ghent. This is a 20-minute film which consists of nothing more than a continuously developing organic image (Figure 3). The picture gets its organic quality by being made up from a myriad of representations of shaded spheres, a technique developed by Norman Badler of Pennsylvania University in order to represent the human body. It is surprising how many forms can be represented in this way. The cover illustration is also by Ken Knowlton and shows the technique used to model human hands.

The technique is best described in a paper by O'Rourke and Badler in *Transactions on Pattern Analysis and Machine Intelligence,* Vol 1, No 3 (July 1979) under the title 'Decomposition of three-dimensional objects into spheres'. The good thing about spheres, from the point of view of computer graphics, is that they project in perspective as discs, so that, with modern raster systems which have hardware disc or circle generation, the drawing of complex objects is very quickly achieved. I have recently programmed the Badler algorithms on a Chromatics graphics system and can vouch for the fact that they are easy to implement and that they work. Try them.