Appears in *Proceedings Graphics Interface '91*, 1991, 224-229.

New Paradigms for Computing in the Nineties

Ronald Baecker
Dynamic Graphics Project
Computer Systems Research Institute and
Department of Computer Science
University of Toronto
6 Kings College Road
Toronto Ontario M5S 1A1 Canada
416-978-6983 FAX: 416-978-4765 E-mail: baecker@dgp.toronto.edu

Abstract

This paper and the accompanying invited talk focus on new paradigms for computing in the nineties, with emphasis on computer supported cooperative work (CSCW). We define CSCW as computer-assisted coordinated activity such as communication and problem solving carried out by a group of collaborating individuals. CSCW represents a paradigm shift for computer science, one in which *human-human* rather than human-machine communications and problem solving are emphasized. This paper describes the nature of work in CSCW, sketches the history of the field, and formulates some issues that are important to ensure progress and future success.

Résumé

Cet article ainsi que l'exposé qui lui est associé font le point sur de nouveaux modèles de calcul pour la décennie actuelle, en particulier dans le domaine du travail coopératif assisté par ordinateur (TCAO). Nous définissons le TCAO comme une activité de coordination assisté par ordinateur, telle que la communication et la résolution de problème dans le cadre d'une collaboration d'individus. TCAO correspond à un changement de paradigms pour l'informatique, pour lequel la communication et la résolution de problèmes homme-homme plutôt q' homme-machine sont mis en relief. Cet article décrit la nature du travail en TCAO, esquisse un historique du domaine, et formule quelques directions importantes pour en assurer l'évolution et le futur succès.

Keywords

Human-computer interaction, user interface design, computer supported cooperative work, groupware, electronic mail, computer conferencing, group decision support systems, teleconferencing, desktop videoconferencing, media spaces.

Paradigms

In his landmark work on the history of science entitled *The Structure of Scientific Revolutions*, Thomas Kuhn (1970) defines *paradigms* as “universally recognized scientific achievements that for a time provide model problems and solutions to a community of practitioners” (Kuhn, 1970, p. viii) He suggests that [the] paradigms “served for a time implicitly to define the legitimate problems and methods of a research field for succeeding generations of practitioners...because they shared two essential characteristics. Their achievement was sufficiently unprecedented to attract an enduring group of adherents away from competing modes of scientific activity. Simultaneously, it was sufficiently open-ended to leave all sorts of problems for the redefined group of practitioners to resolve.” (Kuhn, 1970, p. 10)

Paradigms play a similar role in the engineering and design disciplines, of which the field of “computer science” is one. Paradigms encapsulate breakthrough concepts and technologies which become legitimately fashionable because they suggest and enable new markets and new applications. Although this has been true throughout the history of computing, we shall focus here only on the 80's and the 90's.

Trends in the Eighties

The 80's was an exciting decade in computing. One hardware paradigm became dominant — the personal computer (PC). The personal computer freed users from the tyranny of shared computing under the control of computer centers. As the cost of a reasonably priced machine continued to plummet, it also enabled one to have continuity between a PC work environment at the office and a PC work environment at home.

A number of new interface paradigms contributed to the success of the hardware (Baecker and Buxton, 1987; Baecker, 1987; Laurel, 1990a). WIMP (Window, Icon, Menu, and Pointer) interfaces, pioneered on the Xerox Star and popularized on the Apple Macintosh, empowered millions of new computer users. Non-procedural, graphically-mediated user programming, as pioneered by Visicalc and stabilized through the more enduring technologies of Lotus 1-2-3 macros and HyperCard stacks driven by HyperTalk scripts, gracefully seduced hundreds of thousands of non-programmers into programming without realizing it. The algorithmic bravado of the computer graphics community brought apparent computational photographic realism to all but the most discerning observer.

The achievements of the 80's were in part also due to new software paradigms. The productivity of professional programmers was enhanced through high-level languages exemplified by novel language styles such as concurrent programming, object-oriented programming, and logic programming. Advances in software engineering included sturdier development methodologies and powerful workstation-based programming environments.

Finally, network computing represented an important system paradigm. As computational ambitions and demands on personal computers grew, and as users again realized that no person is an island, the networking of workstations allowed a more congenial sharing of resources than had been possible with earlier minicomputer and time-sharing technology.

Trends in the Nineties

What now for the nineties? Crystal ball-gazing in this industry is perilous, but certain trends seem evident.

One important hardware paradigm for the 90's will be that of *notebook and handheld computers* (Byte, 1991). Miniaturization has advanced to the point that we can today embed a 20 MHZ processor, 4 Meg of RAM, 40 Meg of hard disk, a VGA display, and a digitizing pen in a space no larger that a 8.5" X 11" notepad and in a weight no greater than 3 or 4 pounds. Within the decade these computers will also include speech input and speech and non-speech audio output (Buxton, 1989), and telephone, fax, and photographic interfaces. Alan Kay's dream of the Dynabook (Kay and Goldberg, 1977) will become a reality. The result will be a host of new applications and new users: executives and sales people; artists, architects, and designers; authors and students.

New paradigms often emerge by recognizing that the strength of an existing paradigm is also its weakness. The WIMP interfaces of the 80's will give way to *multi-modal, multi-media* dialogues aided by quasi-intelligent programmed *agents* (Laurel, 1990b). Although reality will not quite match the fantasy of the Knowledge Navigator (Apple, 1988), the quality of the interface will change through the use of handwritten input, gestural input, audio input and output, animation (Baecker and Small, 1990; Baecker, Small, and Mander, 1991), and live and recorded video (CACM, 1989), and through ready access to gigabyte data bases such as research compendia and encyclopedias. The naturalness of direct manipulation will be augmented with the power of simple filing and retrieval agents that will scan data bases, digital archives, and electronic news sources.

Software paradigms for the 90's represent a bridging between the user-programming of personal computer software and the high-level languages of the professional. More powerful user programming and code reuse capabilities will be integrated into comprehensive environments for computer-aided software engineering (CASE). Programming environments increasingly will transact with graphical methods of program representation both in their synthesis, through *visual programming* (Shu, 1988), and in their display and analysis, through *program visualization* (Baecker, 1981; Brown, 1988; Baecker and Marcus, 1989).

Finally, the network computing paradigm of the 80's, driven in part by the merging of computing and telecommunications, is being superceded by a deeper concept of computer supported cooperative work. The remainder of this paper and the bulk of my talk will focus on this new paradigm, and, more generally, on various kinds of computer supported collaborative activities.

Computer Supported Cooperative Work

Computer supported cooperative work (Greif, 1988; Galegher, Kraut, and Egido, 1990) is computer-assisted coordinated activity such as problem solving and communication carried out by a group of collaborating individuals. The multi-user software supporting CSCW systems is known as *groupware*, although the latter term is sometimes broadened to incorporate the styles and practices in group process and interaction that are essential for any collaborative activity to succeed, whether or not it is supported by computer.

Examples of CSCW are now commonplace. The most successful CSCW technology to date has been *electronic mail*. A structured form of electronic mail in which messages are organized by topic and dialogues are often mediated by a convenor is known as *computer conferencing* (Hiltz, 1984). As CSCW is based on the convergence of telecommunications and computation, it can incorporate *teleconferencing*, the act of conferring at a distance with the aid of technologies such as audio and video links. The result is often known as *desktop videoconferencing*. The computer can also be used to facilitate joint problem solving rather than communication *per se*, as, for example, in systems for *collaborative writing or drawing*. If the problem solving is instead directed at issue organization and decision support, it is usually known as a *group decision support system*.

Groupware and CSCW systems thus represent a paradigm shift for computer science, one in which *human-human* rather than human-machine communications and problem solving are emphasized. CSCW systems can integrate voice and video communication with shared digital workspaces, and can support work that occurs both synchronously and asynchronously. Thus CSCW technology enables an expansion of both the concepts of *a meeting* and that of *collaborative work*, allowing participants to transcend the requirements of being in the same place and working together at the same time.

As with so many aspects of modern computing, one of the earliest demonstrations of CSCW occurred in Doug Engelbart's Augmented Knowledge Workshop in the middle 60's (Engelbart and English, 1968). His work included the use of hypertext and hierarchically structured documents which were accessible through shared workspaces and discussable over audio and video links. The 60's and 70's saw the emergence of audio, audiographic, and video teleconferencing (Kelleher and Cross, 1985), technologies which failed to live up to early enthusiasms because they were grounded in part on naive goals of significant travel reductions (Egido, 1990). Canadians active in early research included Herb Bown and Doug O'Brien at the Communications Research Centre of DOC with their concept of a *common visual space* (Sawchuk, et al., 1978) and Gordon Thompson (1975) of Bell Northern Research with his concept of *electronic sidewalks*. The term *groupware* was coined by Peter and Trudy Johnson-Lenz (1980) and became fashionable after it was discovered by the media in the late 80's (Richman, 1987). As has been the case so often with experimental computer science of the past two decades, Xerox PARC was instrumental in sparking the field both with its Colab meeting room (Stefik, et al., 1987) and with the Palo Alto – Portland media link (Abel, 1990). Another major contributing thread was the group decision support system that arose out of the MIS community (Dennis, et al., 1988; Kraemer and King, 1988). Finally, the emergence of regular CSCW conferences in the late 80's (Greif, 1986; Suchman, 1988; Halasz, 1990) seemed to confirm that the field had arrived.

Computer scientists developing groupware need to realize that technical brilliance is not enough. Most groupware applications created to date have failed (Grudin, 1989; Markus and Connolly, 1990), despite having what may appear to be elegant technology. Even more so than with conventional single-user systems, groupware can only be successful insofar as it is responsive to the social and organizational context in which it is embedded. Understanding this context can be aided by insights from a variety of disciplines. The psychology and sociology of groups (McGrath, 1984) can help us understand group processes and interactions. The sociology of organizations informs us about organizational change and technology diffusion. Anthropology contributes the use of conversation analysis and other ethnographic methods (Suchman, 1987). The psychology of media (Short, Williams, and Christie, 1976) teaches us about the behavioural impacts of media and telecommunications. Linguistics is a source of insight into conversations and dialogue structure (Winograd and Flores, 1986).

Yet knowledge of behavioural science by itself does not guarantee successful innovation in CSCW. CSCW development is complex, and requires expertise in many areas of computer science. Human-computer interaction contributes insights into user interface design. Networking and communications teach us about distributed systems. Operating systems and database systems provide useful models of concurrency control. Windowing systems and environments lend us implementation tools. Audio and video technology is required for multi-media aspects of CSCW. Finally, artificial intelligence informs us about the construction of intelligent agents.

Most taxonomies of CSCW technologies distinguish them in terms of their abilities to bridge time and to bridge space. Decision support systems usually involve individuals *working together* in an *electronic meeting room* (Mantei, 1989), mostly working *synchronously*, but sometimes working independently for a while until a facilitator reconvenes group attention and discussion. A number of efforts are directed at groupware for synchronous problem solving by teams of *physically dispersed* individuals. Many of the resulting systems incorporate video and audio links as well as shared digital workspaces, and are now commonly referred to as *media spaces* (Stults, 1986, 1988; Buxton and Moran, 1990; Mantei, et al., 1991). Finally, electronic mail and computer conferencing systems transcend the limitations of both time and space, allowing *asynchronous* communications and problem solving among groups of physically dispersed individuals.

Issues in Computer Supported Cooperative Work

Space precludes an in-depth discussion of the many challenging technical (Ellis, Gibbs, and Rein, 1991) and behavioural (Galegher, Kraut, and Egido, 1990) issues that must be tackled in order to achieve widespread and productive use of such technology. We shall here adopt a somewhat different approach, inspired in part by Ishii

(1990), that focuses on the *seams*, *interfaces*, or *gaps* between different kinds of work. The resulting analysis suggests fruitful directions for research and development.

CSCW systems will be more productive when they bridge the following gaps:

• *The gap between management benefits from CSCW technology and worker benefits from the technology.*

Grudin (1989) has argued, for example, that electronic calendaring systems have failed in great part because management gets the primary benefit while workers assume the primary costs of the time of data entry. One method of avoiding this kind of problem is through *participatory design* (Bjerknes, et al., 1987).

• *The gap between work as computer scientists presume it to be, and work as it really is.*

Many systems built for collaborative writing (Ellis, et al., 1991; Leland, et al., 1988) make strong explicit and implicit assumptions about how people write together. In our own work we postponed system design until we could develop a fairly deep understanding of the realities of collaborative writing. This was based on a literature survey, a series of interviews, and an experiment (Posner, et al., 1991). In other words, builders of CSCW technology need to take seriously the principles of *user-centered system design* (Norman and Draper, 1986).

• *The gap between the use of CSCW technology in vitro and its use in vivo.*

Grudin (1991) asserts that another reason why groupware applications fail is that they are extremely difficult to evaluate. Traditional HCI work has relied both on laboratory experiments (*in vitro*) and field studies (*in vivo*) to evaluate the suitability and effectiveness of new applications and their interfaces. Because groupware requires the coordinated work of a number of individuals (and effective collaboration requires sustained use over a period of time), both laboratory experiments and field studies are difficult to run, and we must be extremely cautious in assuming that results from the laboratory will necessarily apply in the field. (See Baecker and Buxton, 1987, Case Study A, for a good example of this.)

• *The gap between work without computers and work with computers.*

This gap affects all computer use, and not only CSCW. Although a few young computer scientists eschew paper totally, most computer users do some work on paper and some in the computer. Transforming work products from within the computer to paper is straightforward. In most environments, however, moving information from paper or from the visual world into the computer world is difficult. This will change through advances in optical character recognition and through the use of video, but the graceful introduction of these technologies into common work practice is for the most part yet to be achieved.

• *The gap between individual and group work.*

If groupware is to succeed, we need to be able to move smoothly and gracefully between individual and group work. This suggests two more specific gaps that need to be bridged.

• *The gap between work with conventional software and work with groupware.*

The most elegant collaborative editor will be only a toy if there is no clean interface between it and the single-user editors used for most work. At a minimum, there must be a way to export documents back and forth between the two environments. Far better is for the single-user software to be a subset of the groupware system, or for the groupware to supplant the old technology.

• *The gap between work in one's individual office and work in a common meeting space.*

Similarly, one should be able to carry over and use in a common meeting space the tools and resources available in one's office. If the computer environments available in electronic meeting rooms are compatible with and linked to one's personal office computer, part of the problem is solved. One trades access to non-computational resources for the benefits of meeting face-to-face in the common space. Desktop videoconferencing allows one to pick a different point in the cost-benefit space: one remains in his or her own office but accepts the limitations (as well as the strengths) of communicating through a media space in place of the traditional face-to-face meeting.

• *The gap between work in a localized meeting and work in a distributed meeting.*

This and the next gap deal with the pragmatics of using synchronous groupware. There is an enormous difference between using collaborative drawing or decision support tools in an electronic meeting room and using them in a distributed environment across several rooms. Perhaps desktop videoconferencing or video overlays (Ishii, 1990) can help bridge this gap, but much remains to be done.

• *The gap between work across local area networks and work across wide area networks.*

Furthermore, even if technology is successful in use across several rooms connected by a local area network, it may fail when used through a wide area network. Here the developers of asynchronous groupware have an advantage: delays of a few seconds or minutes are usually unnoticeable. The technical and behavioural problems of developing synchronous groupware that is effective across wide area networks are substantial.

• *The gap between synchronous work and asynchronous work.*

Finally, real collaborative work moves smoothly back and forth between synchronous, real-time interaction and asynchronous, off-line activity. Yet most CSCW technology has been developed for one class of work or the other, but not both. To be successful, groupware must allow users fluid motion between working together concurrently and working independently.

Other Computer Supported Collaborative Activities

Our focus on work should not divert us from other rich application domains for computer supported collaboration. Scardamalia, Bereiter, and co-workers (1989), for example, have developed a collaborative hypertext system and method of pedagogy which encourage elementary and high school students to think about and articulate the knowledge they acquire about certain topics. An equally compelling demonstration at the university level has been provided by the Intermedia system (Landow, 1990). Computer supported cooperative education (CSCE) will clearly be an important area for work in the 90's. Anyone who has witnessed the Videoplace work of Myron Krueger (1983), in which he links the body movements of one individual with the hand movements of another in a collaborative video screen dance, will recognize the compelling possibilities of computer supported cooperative play (CSCP). When we think of the prevalence of the video war games of the seventies and eighties, we can only hope that Krueger's work also foreshadows a new collaborative paradigm for adoption by the video games industry.

Acknowledgements

The work of our laboratory is supported in part by the Natural Sciences and Engineering Research Council of Canada, by the Information Technology Research Centre of Ontario, by the Institute for Robotics and Intelligent Systems of Canada, by Apple Computer, Xerox Corporation, and IBM Canada. Their support, and the assistance of the members of the Dynamic Graphics Project laboratory and the Cavecat research team is gratefully acknowledged. Many thanks to Didier Badouel for translating the abstract, and to Beverly Harrison for comments on the manuscript.

References

Abel, M. (1990). Experiences in an Exploratory Distributed Organization. In [Galegher, et al., 1990], pp. 489-510.

Apple Computer (1988). *The Knowledge Navigator*. Video.

Baecker, R. (1981). *Sorting out Sorting.* 30 minute video, Dynamic Graphics Project, University of Toronto. (Distributed by Morgan Kaufmann.)

Baecker, R. (1987). *The Dynamic Image.* 60 minute video, Dynamic Graphics Project, University of Toronto. (Distributed by Morgan Kaufmann.)

Baecker, R. and Buxton, W. (1987). Readings in Human Computer Interaction: A Multidisciplinary Approach. Morgan Kaufmann.

Baecker, R. and Marcus, A. (1989). *Human Factors and Typography for More Readable Programs.* ACM Press, Addison-Wesley.

Baecker, R. and Small, I. (1990). Animation at the Interface. In [Laurel 1990a], pp. 251-267.

Baecker, R., Small, I., and Mander, R. (1991). Bringing Icons to Life, *Proceedings of CHI '91*.

Bjerknes, G., Ehn, P., and Kyng, M. (1987). *Computers and Democracy: A Scandinavian Challenge*. Avebury, Gower Publishing Company.

Brown, M. (1988). *Algorithm Animation*. MIT Press.

Buxton, W. (1989). Introduction to this Special Issue on Non-Speech Audio. *Human Computer Interaction* 4(1), pp. 1-9.

Buxton, W. and Moran, R. (1990). EuroPARC's Integrated Interactive Intermedia Facility (IIIF): Early Experiences. In Gibbs, S. & Verrijn-Stuart, A. (Eds.), *Proceedings of the IFIP WG8.4 Conference on Multi-User Interfaces and Applications*, Elsevier (North-Holland), pp. 11-34.

Byte (1991). Special Issues on Laptop Computers. *Byte* 16(2), February 1991.

CACM (1989). Special Issue on Interactive Technologies. *Communications of the ACM* 32(7), July 1989.

Dennis, A.R., George, J.F., Jessup, L.M., Nunamaker, Jr., J.F., & Vogel, D.R. (1988). Information Technology to Support Meetings. *MIS Quarterly* 12(4), December 1988, 591-624.

Egido, C. (1990). Teleconferencing as a Technology to Support Cooperative Work: Its Possibilities and Limitations. In [Galegher et al. 1990], pp. 351-371.

Ellis, C., Gibbs, S., and Rein, G. (1991). Groupware: Some Issues and Experiences. *Communications of the ACM* 34(1), pp. 39-58.

Engelbart, D. and English, W. (1968). A Research Center for Augmenting Human Intellect. Pro*ceedings of the FJCC* 33(1), pp. 395-410. Reprinted in [Greif 1986].

Galegher, J., Kraut, R., and Egido, C. (Eds.) (1990). *Intellectual Teamwork: Social and Technological Foundations of Cooperative Work*. Lawrence Erlbaum Associates.

Greif, I. (Ed.) (1986). *CSCW 86: Proceedings of the Conference on Computer-Supported Cooperative*

Work. Austin, Texas, Dec. 3-5, 1986. Microelectronics Computer Corporation.

Greif, I. (Ed.) (1988). *Computer-Supported Cooperative Work: A Book of Readings.* Morgan Kaufmann.

Grudin, J. (1989). Why Groupware Applications Fail: Problems in Design and Evaluation. *Office: Technology and People* 4(3), pp. 245-264.

Halasz, F. (Ed.) (1990). *CSCW 90: Proceedings of the Conference on Computer-Supported Cooperative Work*. Los Angeles, Oct. 7-10, 1990. Association for Computing Machinery.

Hiltz, S.R. (1984). *Online Communities: A Case Study of the Office of the Future*. Ablex.

Ishii, H. (1990). TeamWorkStation: Towards a Seamless Shared Workspace. In [Halasz 1990], pp. 13-26.

Johnson-Lenz, P. and T. (1980). Groupware: The Emerging Art of Orchestrating Collective Intelligence. Presented at the First Global Conference on the Future, Toronto, Canada.

Kay, A. and Goldberg, A. (1977). Personal Dynamic Media. *IEEE Computer* 10(3), March 1977, pp. 31-42.

Kelleher, K. and Cross, T.B. (1985). *Teleconferencing: Linking People Together Electronically*. Prentice-Hall.

Kraemer, J.L. & King, J.L. (1988). Computer-Based Systems for Cooperative Work and Group Decision Making. *ACM Computing Surveys* 20(2), June 1988, pp. 115-146.

Krueger, M. (1983). *Artificial Reality*. Addison-Wesley.

Kuhn, T. (1970). *The Structure of Scientific Revolutions*. 2nd Edition. University of Chicago Press.

Landow, G. (1990). Hypertext and Collaborative Work: The Example of Intermedia. In [Galegher et al., 1990], pp. 407-428.

Laurel, B. (Ed.) (1990a) *The Art of Human-Computer Interface Design*. Addison-Wesley.

Laurel, B. (1990b). Interface Agents: Metaphors with Character. In [Laurel 1990a], pp. 355-365.

Leland, M., Fish, R., and Kraut, R. (1988). Collaborative Document Production using Quilt. In [Suchman 1988], pp. 206-215.

Mantei, M. (1989). A Study of Executives Using a Computerized Supported Meeting Environment. *International Journal of Decision Support Systems*, June 1989.

Mantei, M., Baecker, R., Sellen, A.. Buxton, W., Milligan, T., and Wellman, B. (1991). Experiences in the Use of a Media Space. *Proceedings of CHI '91*.

Markus, M.L. and Connolly, T. (1990). Why CSCW Applications Fail: Problems in the Adoption of Interdependent Work Tools. In [Halasz 1990], pp. 371-380.

McGrath, J.E. (1984). *Groups: Interaction and Performance*. Prentice-Hall.

Norman and Draper (1986). *User Centered System Design.* Lawrence Erlbaum Associates.

Posner, I., Baecker, R., and Mantei, M. (1991). How People Write Together. Unpublished manuscript.

Richman, L. (1987). Software Catches the Team Spirit. *Fortune*, June 8, 1987, pp. 125-136.

Sawchuk, W., Bown, H., O'Brien, D., and Thorgeirson, W. (1978). An Interactive Image Communication System Using Narrowband Lines. C*omputers and Graphics* 3(4), pp. 129-134.

Scardamalia, M., Bereiter, C., McLean, R., Swallow, J., and Woodruff, E. (1989). Computer Supported Intentional Learning Environments. *Journal of Educational Computing Research* 5(1), pp. 51-68.

Short, J., Williams, E., and Christie, B. (1976). *The Social Psychology of Telecommunications*. John Wiley & Sons.

Shu, N. (1988). *Visual Programming.* Van Nostrand Reinhold.

Stefik, M., Foster, G., Bobrow, D., Kahn, K., Lanning, S., and Suchman, L. (1987). Beyond the Chalkboard: Computer Support for Collaboration and Problem Solving in Meetings, *Communications of the ACM* 30(1), pp. 32-47.

Stults, R. (1986). Media Space. Xerox Palo Alto Research Centre Technical Report.

Stults, R. (1988). Experimental Uses of Video to Support Design Activities. Xerox Palo Alto Research Centre Technical Report SSL-89-19.

Suchman, L. (1987). *Plans and Situated Action: The Problem of Human-Machine Communication*. Cambridge University Press.

Suchman, L. (Ed.) (1988). *CSCW 88: Proceedings of the Conference on Computer-Supported Cooperative Work*. Portland, Oregon, Sept. 26-29, 1988. Association for Computing Machinery.

Thompson, G. (1975). Technical and Research Aspects of Communication. Appears in *Communications Canada 2000*, University of Toronto Press, pp. 25-49.

Winograd, T. & Flores, F. (1986). *Understanding Computers and Cognition*. Addison-Wesley.