GROUPWARE

Groupware—computer-based applications and features that support group activity—has suddenly become an intense focus of research and development. First becoming a serious possibility with the spread of local area networks, groupware has almost unlimited potential where the Internet and intranets, World Wide Web, mobile computing, and other foundations are in place. Groupware features are being built into operating systems and browsers. But the shift from single-user applications to groupware is not a simple one for designers or users. This entry defines groupware, reviews experiences and lessons learned, and identifies current trends.

When the term *groupware* first appeared in print in 1982, it was used to describe both computer technology that supports groups and the behavioral and organizational effects of introducing such support (1). Today, the term is generally restricted to the technology. Other terms, notably *computer-supported cooperative work* (CSCW), span the technologies and their social and organizational contexts.

GROUPWARE RESEARCH AND DEVELOPMENT CONTEXTS

Each ring in Fig. 1 represents a focus of computer systems development. The outer ring, large systems that serve organizational goals, emerged first with most software developed internally and described as data processing (DP), management information systems (MIS), or information technology (IT). The inner ring, single-user commercial software applications, rode the PC to become a powerful force in the mid-1980s. Research areas focused on individual productivity tools included human factors (HF) and computer and human interaction (CHI, also HCI).

Groupware is usually applied to the two middle rings. PCs were not designed to be networked, and only when significant numbers of them were, in the late 1980s, did commercial software developers expand their horizons from individual productivity tools to small-group support. Much of this was computer-mediated communication (CMC), abetted by telecommunication companies eager to build demand for bandwidth through audio and video technologies. The CSCW conference series, initiated in 1986, became a forum for this work.

Large-group, often project-level, support was approached from a different angle. In the mid-1970s, minicomputer-based office automation (OA) and workstation-based software engineering (SE) addressed problems of communication and coordination at this level. Little progress was made and office automation did not survive as a term or research program, but work on group decision support systems (GDSS, also the more general GSS), often consisting of electronic meeting rooms, has continued, and today's workflow management systems have taken up many of the OA challenges.

This larger context is important because it identifies groupware and the research communities of CSCW and Workflow/GSS as drawing on a range of past and present influences. In the 1980s, computer and software vendor companies focusing attention on supporting networked groups found common interests with researchers and developers oriented toward management information systems, social sciences, and other disciplines. Groupware has a mixed lineage, and CSCW is less a field than a forum that attracts diverse people who have partially overlapping interests and a willingness to overcome the difficulties of multidisciplinary interaction. Recently, in fact, those primarily interested in large-group support have interacted less frequently with those focusing on small-group support, though both identify with the term groupware.

This picture is skewed toward a North American perspective, because that is where the computer, software, and telecommunication companies have exerted the strongest influences, emphasizing small-group support and CMC. A similar picture prevails in Japan and Asia. In Europe, however, there has been a stronger focus on organizational systems and large-group support.

How has groupware fared? Apart from e-mail, progress was disappointing until Lotus Notes and then the World Wide Web took off in the early 1990s. Vendor companies, eyeing the large potential market of small groups, found that the shrink-wrap approach did not work. It was not possible to market groupware the way that word processors, spreadsheets, and games were marketed to individuals. The organizational settings of group activity are too salient to be ignored and too complex to be easily addressed. Large group support has proven equally challenging.

For more on the early history of CSCW and groupware, Grief (2) is a collection of influential papers.

WHAT GROUPWARE IS AND IS NOT

Many writers have struggled to define groupware without improving conceptual clarity. Beyond technology that supports groups, there is an assumption that all or most group members participate directly in using the application. Ordinary multi-user databases are usually not included; they create the illusion of being a single-user application, they do not include or foster a sense of the group. Broadcast technologies such as the multicast backbone (MBONE) and even point-to-point

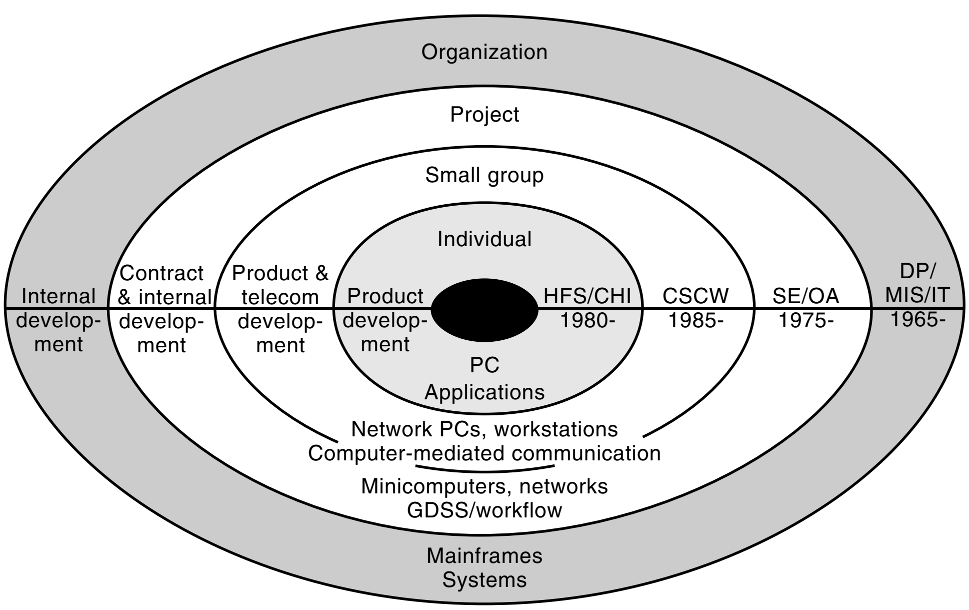

Figure 1. Groupware research and development contexts.

technologies such as e-mail are sometimes disputed, although once distribution lists are used e-mail certainly becomes a group tool. Other technologies, such as computer assisted design/computer assisted manufacturing (CAD/CAM), do support groups, yet are rarely called groupware. Baecker (3), the most comprehensive collection of readings on groupware and CSCW with more than 70 papers, reflects the literature at the time it was compiled, but contains nothing on computer-mediated education and distance learning, project-level software engineering support, workflow management, computer-integrated manufacturing, and other topics.

Bannon and Schmidt (4) and others have made a case for establishing a field or discipline, but today we see more of a forum, an undisciplined marketplace of ideas, observations, issues, and technologies. Differences in interests and priorities are as notable as the shared interests. People arrive from and return to different places. Not everyone speaks the same technical language or makes the same assumptions. To understand what is encountered or read concerning groupware, it is essential to reflect on this muddy heritage, otherwise frequent misunderstanding and the lack of intellectual coherence is frustrating. But when understood and respected, the differences form the core of richer, shared understandings.

Grudin (5) contains more detail on the origin and composition of those working on groupware and involved in CSCW research.

THE SIGNIFICANCE OF GROUP SIZE

The most dramatic shift for many developers was going from a single-user application to groupware. Early word processors, spreadsheets, and other individual productivity tools could be designed with a focus on the perception, cognition, and actions of an individual. The social context of the activity could be ignored. As product developers extended their view to computer support for groups, many confronted social issues in customer settings—group dynamics—for the first time. With groupware, social, motivational, and political aspects of workplaces become crucial (6).

We have further distinguished among support for small-groups, large groups, and entire organizations. Many desktop conferencing systems with audio or video links work best with two to five simultaneous participants. Other applications, such as electronic meeting rooms, require six or more participants to be useful. Still others, such as newsgroups, may not survive unless they quickly obtain one hundred or more subscribers.

A deeper analysis reveals patterns in the applications that serve groups of different sizes. These patterns explain the differing assumptions, motivations, and priorities of the developers and users of different kinds of groupware.

People often work together in small groups or teams because they have a high need to communicate. Small-group support, such as desktop videoconferencing or voice annotation, is typically focused on *communication.* As the group size increases, there is an increasing need to focus on *coordination* of work between subgroups (7). The co-authoring tool needed by two scientists writing a paper together will be different from that needed by twenty writers laboring on different parts of a major piece of documentation.

The members of a small group typically have a high degree of shared purpose. A designer of groupware to support them can assume *cooperation* among users. In contrast, different people in an organization or very large group will inevitably find themselves in *conflict.* A tool that requires full cooperation may well fail in large group settings; for example, one group may not enter information into the system if it will be visible to a competing group.

Applications to support small groups often find themselves in a competitive market, and must enhance the *human–computer interface* to get an edge over similar products. A very large system, developed or customized for one organization, must stress *functionality,* and may have fewer resources for enhancing the interface for the one group.

Groupware developers and users with a focus on communication, cooperation, and the interface may seem to be out of touch to those whose focus is coordination, conflict management, and functionality—and vice versa. Worse, systems built on one set of assumptions that must work in the other context are likely to fail. It may be misleading to use one term, groupware, to span these different sets of requirements.

GROUPWARE TYPOLOGIES

Categorization by Group Activity

Size is only one of many distinctions proposed in typologies or categorizations of groupware. Figure 2 presents a variant of a widely used space and time categorization introduced and refined by DeSanctis and Gallupe (8) and Johansen (9). Representative applications illustrate the different cells. Activity can be carried out in a single place (top row), in several places that are known to the participants, as in electronic mail exchanges, for example (middle row), or in numerous places not all of which are known to participants, as in a message posted to a netnews group (bottow row). Activity can be carried out

Place \ Time	Same	Different but predictable	Different but unpredictable
Same	Meeting facilitation	Work shifts	Team rooms
Different but predictable	Teleconferencing Videoconferencing Desktop conferencing	Electronic mail	Collaborative writing
Different but unpredictable	Interactive multicast seminars	Newsgroups	Workflow

Figure 2. A 3 × 3 map of groupware options.

in real time; that is, in one unbroken interval, as in a meeting (left column). Alternatively it can be carried out at different times that are highly predictable or constrained, as when you send mail to a colleague expecting it to be read within a day or so (middle column). Or it can be carried out at different times that are unpredictable, as in an open-ended collaborative writing project (right column).

This easy-to-learn typology facilitates communication and is widely used, but not without risk. Most of us, as we go about our work, engage in some face-to-face meetings and some distributed and asynchronous communication. Our work involves both communication and coordination. Technology designed to support activity in one cell can fail by negatively impacting activity in another. For example, a stand-alone meeting support system that provides no access to other on-line materials may be useless. Noting the interdependencies among activities, Robert Johansen (9) calls for "any time, any place" support.

Categorization by Type of Task

Work typically involves communication among participants, collaboration or cooperation in a shared information space, and coordination of contributions. The technology features that support these tasks are the essence of groupware, whether these features are found in a groupware product or are integrated into other products, such as office systems.

Groupware communication features enable people to communicate with one another. The communication may be real-time as in video conferencing, voice conferencing, and text-based chat sessions. The communication may be an asynchronous electronic mail message but still contain video, voice, text, and other media.

Shared-information-space features provide virtual places where people create and manipulate information. These features often include a shared repository to store and retrieve information. Like the communication features, these may be real-time or asynchronous. Real-time features are found in multi-user white boards and application-sharing in desktop conferencing systems, brainstorming tools in meeting facilitation systems, and multi-user virtual worlds. Asynchronous features include information management, document management, multi-user hypertext systems, and threaded discussions. Information retrieval features such as hypertext links, navigational views, and full-text search support retrieval from shared information spaces.

Coordination features facilitate interactions between or among participants. Virtually any collaborative activity requires some degree of coordination, and most groupware products include some sort of coordination features. For example, real-time communication features such as video conferencing are necessarily coupled with coordination features for establishing communication channels among the participants. Coordination features are essential when interacting asynchronously in shared information spaces. Access control features limit who can participate in a shared space. Library features in document management systems include checking out documents for revision and maintenance of document versions. These features coordinate interactions at a relatively fine-grained level, and aim to do it as unobtrusively as possible.

Some technologies support coordination at a more macroscopic level, facilitating management of the overall flow of work. These technologies include calendar and scheduling, project management, and workflow management systems.

Categorization by Groupware Technology

Groupware technologies generally combine communication, information-sharing, and coordination features. Often, however, features from one category dominate, and these dominant features can serve to categorize groupware products and prototypes. For example, electronic mail and video conferencing products predominately serve interpersonal communication; document management products predominately provide a shared information space; and workflow management systems predominately coordinate the flow of work.

The next three sections describe technologies from each of these three categories, identifying where these technologies use features from the other categories.

COMMUNICATION TECHNOLOGIES

Electronic Mail

Electronic mail or e-mail is the most successful, best-known groupware technology. It is also a key element of well known groupware products such as Lotus Notes, Microsoft Exchange, Novell Groupwise XTD, and ICL Teamware. After decades of use and widespread acceptance, electronic mail is a relatively mature groupware technology. It continues to evolve, however, to meet evolving capabilities of computers and users' changing expectations.

E-mail is inherently structured. Messages consist of a series of field labels (To, From, Subject, etc.) and field values, ending with a body field containing the content of the message. An important step in the evolution of e-mail was to provide a capability for creating additional fields. The Information Lens (10) demonstrated how these fields, combined with agent technology, could help users process and handle their mail. Today many groupware products, including most e-mail systems, contain tools for constructing such agents, and improved human-computer interfaces that make them more usable. Borenstein (11) proposed a significant further step in which programs (similar to Java) are embedded within e-mail messages and executed by the recipient.

Until recently, e-mail systems used either time-sharing architectures with poor performance and usability, or file server architectures with poor reliability and scalability. The current generation of e-mail systems (characterized by Lotus Notes, Microsoft Exchange, and Novell Groupwise XTD, among others) have adopted client-server architectures. These systems can serve as universal "in-boxes" for e-mail, voice mail, fax, and video messages. Experience with the Pandora Multimedia System, a research prototype developed at Olivetti Research Labs, showed that video mail can be a popular feature (12).

Real-Time Conferencing

Viewed from a computing perspective, the ubiquitous telephone combines simple, inexpensive client hardware with a powerful network and server infrastructure. Emerging computer-based communication technology may soon replace the telephone in many settings by offering greater capability and flexibility at lower cost. The current generation of personal computers has audio capabilities surpassing those of the telephone handset, supports live video, and can assume some of the processing performed centrally by telephone companies. Both intranets and the Internet can replace the telephone infrastructure as the network for voice communication. Existing software supports voice communication between any two computers connected to the Internet at no cost. Real-time video communication is also possible over phone lines, ISDN (integrated services digital network) lines, and ethernet.

Today's desktop video conferencing systems enable people to see small, low-resolution pictures of one another while conversing. A video camera mounted on or near the display transmits a video (and audio) signal, which appears in windows on other participants' displays. Advances in camera technology, compression algorithms, and network technology are rapidly improving the performance and driving down the cost of video conferencing. Performance has not reached television quality; most systems can maintain a maximum of about 12 to 15 frames per second. Nonetheless, the market and the number of vendors for this technology are expanding rapidly; Perey (13) lists 40 vendors of desktop video conferencing systems.

Problems frequently reported with desktop video conferences are:

1. Difficulty of making eye contact,
2. Insufficient resolution to recognize important visual cues,
3. Lack of appeal of static "talking heads"

Considerable effort has been directed at these problems. Hydra (14) consists of a set of small units, each containing a camera, microphone, monitor, and speaker. Up to four people at different locations could meet using Hydra as though seated around a table. At each location, three Hydra units are distributed around a table to represent the other three participants. When a meeting participant turns to look at the person on one monitor, everyone can see and interpret this shift of attention.

The miniature units of Hydra, with camera close to monitor, created an impression of eye contact. The MAJIC system enables eye contact with life size images of participants (15–17). Not a desktop system, MAJIC's key feature is a large screen that is transparent from one side but reflective on the other side. The display image is projected on the reflected side, and a camera captures the participant's image from the other side. It is easy to establish eye contact and recognize nonverbal cues such as gestures or changes in body position.

In an interesting, innovative project, Inoue et al. (18) examined the way television producers vary camera shots, in an effort to automatically produce a more interesting mix of images in video conferences.

Some researchers have questioned the value of video in interpersonal communication. Summarizing the results of many researchers, Whittaker (19) notes that speech is the critical medium for interpersonal communications, and video can do little more than transmit social cues and affective information. Video adds value when used to show physical objects, not speakers and audiences. Heath, Luff, and Sellen (20) similarly conclude that "the principle concern in media space research with supporting (mediated) face-to-face communication has inadvertently undermined its ability to reliably support collaborative work." They observe that "where individuals do, for example, try to write a paper together using the media space, or provide advice on the use of new software, the inability to see and share objects and shift one's views of each other causes frustration and difficulty for those involved." Experiments by Williams (21) found more use of video among speakers of mixed linguistic background in conflict situations; added value of video in mixed language settings is also reported by Gary Olson (personal communication).

Multicast Video and Audio

Multicast technologies broadcast information, and these broadcasts can be received by a potentially large audience. The Multicast Backbone (MBONE) on the Internet (22) distributes live audio and video presentations. Many special interest groups within the Internet community have conducted online conferences using MBONE coupled with a shared white board program to display presentation materials.

Isaacs and her colleagues at SunSoft (23,24) developed and evaluated a system called Forum that uses advanced MBONE technology to broadcast audio, video, and slides to a live audience. The speaker uses Forum to present and annotate slides, identify and conduct polls of the audience, and call on audience members. Audience members view a video image of the speaker, respond to polls, and request permission to speak in one window. In a second window audience members view the slides, and in a third window they can view a list of all audience members and exchange private messages.

A controlled study of face-to-face and distributed presentations (24) found that more people attended Forum presentations, but they paid less attention than face-to-face audiences, simultaneously reading their mail or talking to co-workers. Audiences strongly preferred attending Forum presentations over face-to-face presentations, but the speakers, not surprisingly, preferred the interactivity and feedback of the face-to-face presentations.

SHARED INFORMATION SPACE TECHNOLOGIES

Real-Time Shared Spaces

Real-time shared information spaces enable people to work together synchronously with awareness of other participants and their activities. Multi-user white boards and other multi-user applications enable teams to draw or type concurrently in a shared space. Meeting facilitation systems provide shared spaces for capturing and manipulating the contributions of all meeting participants. MultiUser Dungeons (MUDs), and virtual worlds create the experience of interacting with people in an artificial environment.

Shared White Boards and Application Sharing. *Shared white boards* and *application sharing* are two features of desktop conferencing technologies. Shared white boards are simply multi-user graphics editors. In general, all users can draw, type, or telepoint simultaneously on the same virtual white board, can import images from other applications, and can store images generated in advance for a "group slide show." These objects often serve as conversational props (25).

Application-sharing technologies allow a group to work together using a single-user application running on one of their computers. The software transmits the application's windows to all users and integrates all users' inputs into a single input stream. Examples include HP's SharedX, X/TeleScreen, Smart200, Fujitsu's DeskTopConferencing (DTC), and Microsoft's NetMeeting.

Video conferencing and multi-user applications usually run in distinct windows that compete for display space. The video cannot provide information about gestures or direction of gaze that would communicate which objects people are attending to within the shared application. ClearBoard (26) solves this problem by integrating the video image of participants and the shared information space. The conceptual model for ClearBoard was working on opposite sides of a clear sheet of glass. ClearBoard overlays a video image with a multi-user application to achieve the same effect, reversing the image to achieve the same left-right orientation.

Architecturally, desktop conferencing systems differ as to whether the application is centralized or replicated (27). Both architectures feature a conference agent, the core of the conferencing product, running on all participating computers. However, the method by which a conference agent establishes and manages communication in a desktop conferencing session differs across the architectures.

The centralized architecture is the foundation for shared-application technologies. A conference agent intervenes in the communication between a single-user application and the computer's window system. The application's outputs are captured by the conference agent and transmitted to the conference agents on all participating computers. These agents convey the output to the window systems, which present it to the users. A user at any computer may interact with the application's objects using keyboard and mouse. The conference agent integrates these inputs and delivers a coherent input stream to the application. To achieve a coherent input stream, the conference agent generally enforces a *floor control* policy, accepting inputs from only one user at a time.

The replicated architecture is the foundation for most shared white boards and other multi-user applications. The same application runs on each computer, and the conference agent tries to ensure that all copies of the application remain synchronized. The conference agents do not transmit application output to other computers. Instead, they ensure that all users' inputs are distributed simultaneously to all copies of the application. As before, the conference agent enforces floor control policies. But with this architecture the policy may also permit simultaneous interactions with application objects.

Meeting Facilitation. University management science departments have long studied business meetings and sought ways to improve meetings. Their research has led to development of technologies, including hardware, software, and techniques for improving meetings. Technologies such as GDSS and GSS, previously mentioned, are in use today.

Researchers at the University of Minnesota developed SAMM (Software-Aided Meeting Manager) as an integrated suite of tools intended to support meeting processes such as issue identification, brainstorming, voting, and agenda management (28). This technology builds on a research program defined in Ref. (8) that integrates behavioral science, group process theory, and adaptive structuration theory.

Jay Nunamaker and his colleagues at the University of Arizona developed similar meeting facilitation prototypes, which Ventana Corporation integrated into a commercial product called GroupSystems (29) and IBM marketed as TeamFocus (30). The activities supported by GroupSystems included exploration and idea generation, idea organization and categorization, prioritizing and voting, and policy development and evaluation. Several different tools may support each of these activities. As a meeting evolves, a human facilitator selects tools to support the current processes.

Support for face-to-face meetings remains an active area of CSCW research for technology developers as well as social scientists. For example, Streitz et al. (31) developed a system called DOLPHIN that includes a large, interactive electronic white board and individual workstations for meeting participants. The design of DOLPHIN was based on observational studies of editorial board meetings where an electronic newspaper was planned and created. Using DOLPHIN, board members can create and share informal information such as freehand drawings or handwritten scribbles, and formally structured information such as hypermedia documents. Mark et al. (32) report that groups organized more deeply elaborated networks of ideas using DOLPHIN.

MUDs, MOOs, and Virtual Worlds. MultiUser Dungeons (MUDs) and their object-oriented extensions (MOOs) are multi-user text-based, virtual worlds. (The term *dungeon* has become a bit of an embarrassment, so the D is often rechristened *Dimensions* or some other word.) MUDs maintain information about users, objects, and interconnected rooms. The MUD users interact with this database, moving from room to room, manipulating objects, and communicating with other users. The interconnected rooms form a virtual world described in text. Users type simple commands such as "Go north" to move from one room to another. When a user enters a room the MUD displays its description, including any objects or other people in the room. Users in the same room can talk to one another and interact with the objects.

A MOO includes object-oriented tools for extending the MUD by building new objects and rooms. The heart of a MOO is a shared information space which supports communication. Curtis and his colleagues (33) describe extensions including windows-based user interfaces, shared tool access, audio, and video. When a user "looks at" a map found in a MOO, a window could open that shows the map. Technical implementations of these extensions are described in (34,35).

The emergence of the virtual reality modeling language (VRML) standard has allowed evolution from text-based MUDs and MOOs to graphical, three-dimensional virtual worlds. In these worlds, participants are represented by graphical *avatars.* Damer, Kekenes, and Hoffman (36) evaluated five prototypes that provide multi-user graphical virtual realities. Participants communicate through text-based chat windows, as in MUDs. Greenhalgh and Benford (37) developed and tested a virtually reality teleconferencing system called MASSIVE that enables a group to interact using audio, graphical, and textual media. Bowers, Pycock, and O'Brien (38) studied social interactions during a MASSIVE virtual meeting and identified problems in turn taking and participation that must be addressed for this technology to be widely accepted.

Asynchronous Shared Spaces

This section describes three technologies for storing and organizing information. Asynchronous computer conferencing tools organize information around ad hoc topics. Document management systems are specialized for supporting the creation and maintenance of electronic documents. Information management tools provide flexible frameworks for diverse information structures.

Asynchronous Computer Conferencing. *Asynchronous computer conferencing* is among the oldest forms of groupware and continues to be widely used under such labels as *bulletin boards, threaded discussions, news groups,* and *public folders.* These technologies provide shared shared information spaces which are typically organized around interest areas. Computer conferencing technology maintains databases of messages organized as collections of tree structures. The starting message is the head of a tree and responses to it are branches. Conferencing clients typically display the tree structure so that users can follow the thread of a discussion.

The topic-and-response tree structure inherent in computer conferencing is widely used in groupware systems. The first version of Lotus Notes was a computer conferencing system with support for both wide and local area networks, and Notes databases still support the conferencing organizational model. Other groupware products that support asynchronous computer conferencing include Netscape's CollabraShare and Attachmate's OpenMind.

Document Management. Document management systems complement and are integrated with word processors, publishing systems, and other media editors. Instead of storing and retrieving documents in a file on a local disk or file server, documents are stored on and retrieved from a document management server. The basic elements of a document management system are a repository for the document objects, a database of meta-information about the objects, and a set of services.

The essential document management services are access control, concurrency control, and version control. Access control determines who can create, modify, and read documents. Concurrency control, preventing different authors from changing the same document at the same time, is generally accomplished by *checking out* the document to the first person who requests write access. Other users can read or copy the document but cannot edit it.

Version control determines whether a modified document replaces the original or is saved as a new version and how long old versions are retained.

Document management systems rarely maintain information about the semantics or structure of the documents they manage. Whether text, graphics, video, or a CAD drawing, to the system it is a blob of unknown content. The semantic information, essential for managing and finding documents, is included in the document meta-information. This database includes the author, date, version number, check-out status, and access permissions. It may also include user-supplied keywords, application-specific fields, position within a hierarchy of folders, and relationships to other documents. A user can, for example, search for all documents written by a certain author between two specified dates.

The World Wide Web offers an ideal environment for document management services. Few web servers, with the notable exception of Hyper-G (or HyperWave), provide these services yet, but vendors are integrating web technology and document management systems. Hyper-G is a web server with integrated access control and sophisticated information retrieval capabilities, including the ability to navigate through a three-dimensional representation of the document space (39,40).

Information Management. Information management technologies such as Lotus Notes combine features of a document management system with structured objects. Most document management systems treat documents as uninterpretable; information management systems, in contrast, manage the structure of the document objects. Lotus Notes represents documents as a collection of named fields and their values. Some fields may contain text, graphics, video, audio, or other media. Other fields contain predefined keywords, dates and times, or other structured data that either the computer or a person can interpret. The combination of structured and unstructured fields constitute a semistructured document.

With Information Lens, Malone et al. (41) established the power of semistructured documents as a foundation for collaborative work. A research prototype called Oval (42) demonstrated that semistructured documents can contribute to radically tailorable tools for collaborative work. Oval could be customized to behave similarly to gIBIS (43), The Coordinator, Lotus Notes, or Information Lens. The current version of Lotus Notes integrates the basic features of Oval to create a rapid application development environment for workgroup applications.

Hypertext provides an alternative way of organizing information elements. SEPIA (44) is a hypertext authoring system that links nodes within activity spaces. These spaces are designed to support the tasks of content generation and structuring, planning, arguing, and writing under a rhetorical perspective. An interesting feature of SEPIA is its support for multiple modes of collaboration. A graphical browser reveals to authors working within the same composite node which component node each person has checked out. Aware of working in the same space, they have the option of entering a tightly-coupled collaborative mode by launching a desktop conferencing tool.

COORDINATION TECHNOLOGIES

Calendars and Scheduling

Calendar and scheduling products often serve as personal information management systems while helping teams coordinate their work. Individual users are supported by personal calendars, action item lists, contacts lists, and other features. Coordination is supported by group calendars, meeting reminders, on-line rolodexes, and especially by scheduling functions that aid in searching the calendars of multiple users to find convenient times for meetings and schedule resources such as meeting rooms. Integration with e-mail can facilitate the invitation process.

Support for meeting scheduling has been an active research area for over a decade; in fact, it has been adopted by the distributed artificial intelligence community as a demonstration problem on which to test approaches. Nevertheless, scheduling features in commercial products went unused for many years due to the lack of a critical mass of use in most environments—too many people found paper calendars more convenient (6,45). Calendar applications have matured, sporting better interfaces, a range of individual-support features, and e-mail integration. Users and technical infrastructures have also matured, leading to widespread use of scheduling in some environments (46).

Workflow Management

Workflow management systems provide tools for coordinating work by managing the task sequence and the flow of information and responsibility. Workflow management technologies were first created to support imaging applications such as insurance forms processing. To improve efficiency and accountability, insurance companies installed technology to scan paper forms and process the form images. Workflow applications were developed to route information from one person to another when each task was completed.

Figure 3 shows a reference architecture for workflow management systems developed by the Workflow Management Coalition. The central component of this architecture, the workflow enactment engine, controls the flow of work in accordance with a stored model of the work processes. This model is created using the system's Process Definition Tools. A model describes the tasks, their sequence, the flow of data, applications used to perform the tasks, and the roles taken by people and systems in performing the work. Some systems include simulation and analysis tools that predict performance of the modeled work process and identify potential problems or errors in the model.

New work is initiated and delivered to the responsible users through the Worklist Tool: When a user selects a task from the Worklist, the Workflow Enactment Engine may invoke applications needed to support the user's performance of the task. Administration and monitoring tools provide information about the performance and cost of a workflow and support dealing with exceptional conditions that were not included in the model.

The Process Definition Tools component is of special interest because its user models the workgroup processes. Until recently, system definers described workflow models as a list

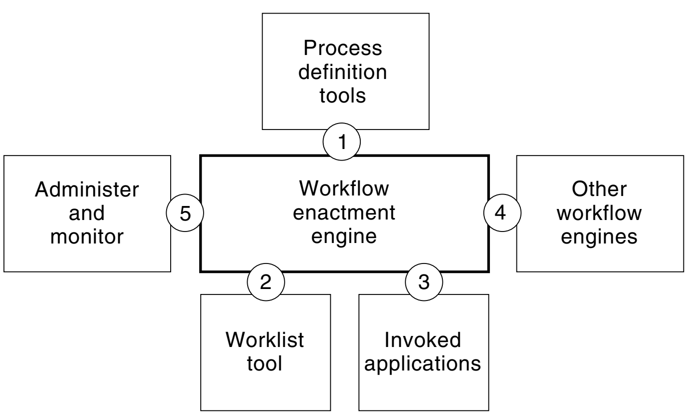

Figure 3. The workflow management coalition reference architecture.

of preconditions and tasks, but most systems today offer a graphical editor for defining the process flow. Most systems adopt an input process output (IPO) model, but an exception is the Action Workflow model (47).

IPO models originated in process analysis and computer programming flowcharts. Their principal advantage is that they are conceptually easy to understand. A disadvantage is that they encourage an oversimplified, unidirectional, sequential view of business processes. The waterfall model would be a natural outcome of using IPO models to describe software engineering practices. An example of an IPO modeling method is information control net (ICN) developed by Ellis (48). The syntactic elements of ICN and a simple ICN model are depicted in Fig. 4.

The Action Workflow model is more difficult for a novice to interpret. Business processes are represented as cycles of communication acts between a customer and a performer. In the simplest cycle the customer requests a deliverable, the performer agrees to produce it, later the performer reports its completion, and finally the customer agrees that the deliverable meets its requirements. An Action Workflow system supports communication about the work among all participants. Of course, each of the four basic communication acts can require additional communication, represented as additional cycles. An example of a very simple Action Workflow model is shown in Fig. 5.

CHALLENGES TO GROUPWARE DEVELOPMENT AND USE

Technical Challenges

Groupware development faces many technical challenges, few of them unique. More efficient compression algorithms, faster processors, satellite communications—everything contributes to improvements. We will restrict ourselves to a few technical problems that are driven in part by the nature of groupware.

Integration of media is an unfinished trend. Many groupware successes come from integrating technologies that previously existed in isolation. Lotus Notes integrated e-mail and information sharing; modern meeting schedulers integrate calendars with e-mail.

Interoperability is a key to supporting group use in heterogeneous environments. Much groupware must be accessible to most group members. If people use incompatible calendars, scheduling features go unused. Groupware applications that must work in concert with other software rely on technical standards. It is futile to develop a coauthoring tool if it entails building a new full-function word processor; on the other hand, a standard interface to an existing word processor provides an opportunity.

Insufficient flexibility is a major problem for groupware. A technical solution may be "reflective systems" that contain modifiable representations of their own behavior (49). Dourish (50) addresses another technical issue: graceful conflict resolution that may enable parallel activity.

Social and Organizational Challenges

Groupware failures far outnumber successes. The following list draws on the account of non-technical challenges to designing, developing, and deploying groupware in (6).

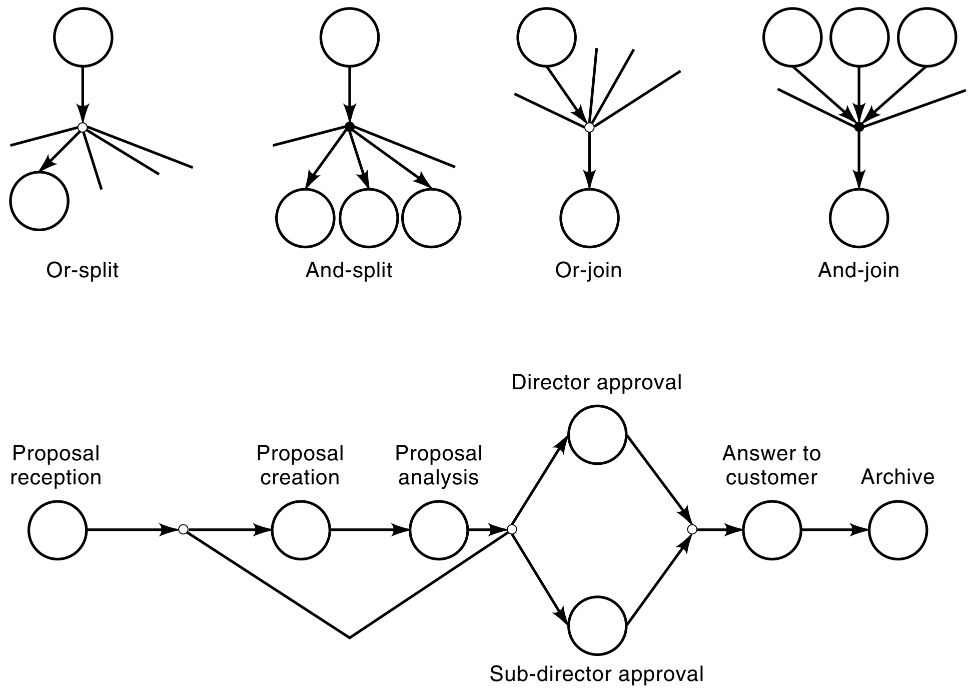

Figure 4. The syntactic elements of information control networks and a simple example of a workflow model composed using these elements.

1. Disparity in work and benefit. Groupware applications often require that some people do additional work. Often they are not the primary beneficiaries, and may not perceive a directed benefit from complying.
2. Critical mass, Prisoner's dilemma, and the Tragedy of the Commons problems. Even in situations where everyone would benefit, groupware may not enlist the critical mass of users required to be useful. Alternatively, it can fail because it is never to any one individual's advantage to use it: the prisoner's dilemma. Markus and Connolly (51) detail these problems. The Tragedy of the Commons describes a situation where everyone benefits until too many people use it. This can be a problem for highways and, perhaps, information highways.
3. Disruption of social processes. Groupware can lead to activity that violates social taboos, threatens existing political structures, or otherwise demotivates users crucial to its success. Much of our knowledge of social conventions is implicit and cannot be built into today's systems.
4. Exception handling. Groupware may not accommodate the wide range of exception handling and improvisation that characterizes much group activity. The significance of this for information systems has been demonstrated by detailed ethnographic studies (e.g., 52,53).
5. Unobtrusive accessibility. Features that support group processes are used relatively infrequently, requiring unobtrusive accessibility and integration with more heavily used features.
6. Difficulty of evaluation. The almost insurmountable obstacles to meaningful, generalizable analysis and evaluation of groupware deter learning from experience.
7. Failure of intuition. Intuitions in research, development, and use environments are especially poor for multi-user applications, resulting in bad management decisions and error-prone design processes. Certain technologies, particularly those that might benefit managers, tend to be viewed too optimistically; the value of other technologies is overlooked.
8. The adoption process. Groupware requires more careful implementation or introduction in the workplace than product developers have recognized. Shrinkwrap groupware is impractical.

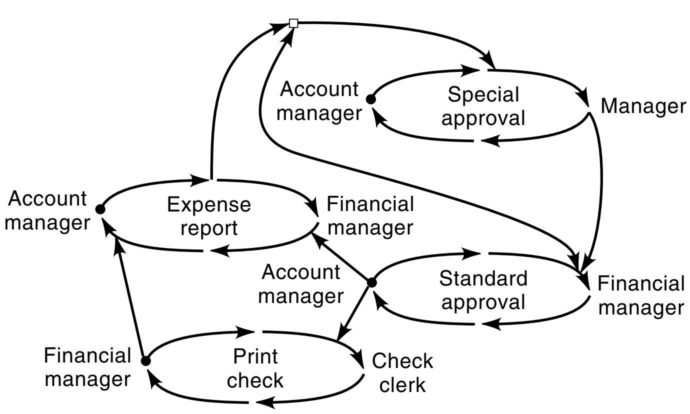

Figure 5. A simple workflow model using the action workflow modeling approach.

NEW APPROACHES TO UNDERSTANDING REQUIREMENTS

Market research and consultants are of uncertain help with groupware, at least until we have more experience. Their approaches work better for assessing individual preferences. Another traditional approach, hiring a domain expert, is highly susceptible to the individual's biases.

Good interactive software design practice does provide a foundation. Gould (54) summarizes techniques from the field of human-computer interaction, focusing on early and continual user involvement and user examination of prototypes, iterative design, and consideration of all aspects of usability in parallel. The use of these techniques in organizational settings is summarized in (55,56).

Participatory design approaches have also been refined for decades, in particular sociotechnical design from England and collaborative Scandinavian approaches (57–59). These approaches maximize user involvement in development, focusing on techniques communication, education, and contribution. Initially used primarily on large systems, they are being adapted to groupware development.

Contextual inquiry, analysis and design is a powerful approach honed by Beyer and Holtzblatt (60) to capture and apply the best findings of field studies in a rapid manner. It centers on interviews conducted as work is in progress, an intrusive but efficient method for gathering data, with the goal of establishing a shared understanding of interviewer and worker about the work practice. The data from a series of interviews is then rigorously analyzed to reach an understanding of the work context and practice in a form that can be communicated to other design team members.

The IT field contributes, drawing on social science and management studies. Galegher et al. (61) is a compendium of social science work. Orlikowski (62) conducted an influential study of the introduction of Lotus Notes in a consulting organization. She found that the reward structure greatly affected the reception of the technology by differentially affecting willingness to share information.

Ethnographic or anthropological studies are labor-intensive but can provide detailed knowledge and on occasion general insights into group and organizational behavior. Perin (63) showed the mixed benefits and costs of e-mail in organizational settings, perhaps explaining its cautious spread over the first twenty years. Bowers et al. (53) described work processes in a print shop that adopted a workflow management system, revealing that flexibility in the service of exception-handling was obstructed by systems that were based on notions of standard processes.

FUTURE DIRECTIONS

The tumultuous arrival of the World Wide Web demonstrates the futility of trying to forecast the future, but some directions seem clear. We can confidently anticipate the increasing integration of media. Standards and interoperability will continue to drive progress, providing substantial short-term benefits with perhaps some long-term inefficiency. Technologies will be adopted enterprise-wide, rather than group by group, and most groupware will be integrated with e-mail, intranets and the Internet. Increased attention by designers and developers to social issues and group dynamics will spur research into organizational and group behavior.

Challenges remain. In order to incorporate greater understanding of work processes into a system, and thus better support the processes, it is tempting to consult a standard policy manual, the official procedures for conducting work in an organization. However, social scientists have noted that standard procedures are often not a prescription. They may represent a goal to strive for, the external face a company wishes to present, or a way to allocate responsibility for a breakdown, in full awareness that corners have to be cut. Work to rule as industrial sabotage reflects our awareness that the rules are neither efficient nor generally followed.

Studies reveal that the reality of work practices is more chaotic than is generally recognized. The orderly face presented to the outside world often masks a far less orderly internal operation. This is a challenge to groupware developers and users.

The rapid development of the communication technologies such as desktop videoconferencing, and information sharing technologies such as the World Wide Web and Lotus Notes, is transforming the computer from a computing machine on our desk to a window onto the world. The window is not perfectly transparent—it filters, it selects—but we are moving toward greater transparency.

This has many benefits. But one side effect, sure to be disruptive in the short term, is that the window will reveal the underlying chaos or nonroutine activity that Suchman and others report. The masks and myths of smooth, consistent operation are being stripped away. Few people are aware of the extent of disorder that exists—our memory and our customs suppress awareness of it. Revelation—seeing the violations, the irregularities, the inconsistencies—will often be highly unsettling.

Possibly we can use technology to recreate the masks and the myths. Perhaps the new technologies will be suppressed. If not, their use will surely lead to the rapid evolution of new social practices and organizations.

BIBLIOGRAPHY

1. P. Johnson-Lenz and T. Johnson-Lenz, Groupware: The process and impacts of design choices, in E. B. Kerr and S. R. Hiltz (eds.), *Computer-Mediated Communication Systems: Status and Evaluation,* New York: Academic Press, 1982, pp. 45–55.
2. I. Greif (ed.), *Computer-Supported Cooperative Work: A Book of Readings,* San Mateo, CA: Morgan Kaufmann, 1988.
3. R. M. Baecker (ed.), *Readings in Groupware and Computer-Supported Cooperative Work,* San Mateo: Morgan Kaufmann, 1993.
4. L. Bannon and K. Schmidt, CSCW: Four characters in search of a context, in J. M. Bowers and S. D. Benford (eds.), *Studies in Computer Supported Cooperative Work: Theory, Practice, and Design,* Amsterdam: North-Holland, 1991, pp. 3–16. Reprinted in (3).
5. J. Grudin, CSCW: History and focus, *IEEE Comput.,* **27** (5): 19–26, 1994.
6. J. Grudin, Groupware and social dynamics: Eight challenges for developers, *Commun. ACM,* **37** (1): 92–105, 1994. Republished in R. M. Baecker et al. (eds.), *Readings in Human-Computer Interaction: Toward the Year 2000,* San Mateo, CA: Morgan Kaufmann, 1995, pp. 762–774.
7. T. W. Malone and K. Crowston, The interdisciplinary study of coordination, *ACM Computing Surveys,* **26** (1): 87–119, 1995.
8. G. L. DeSanctis and R. B. Gallupe, A foundation for the study of group decision support systems, *Manag. Sci.,* **33** (5): 589–609, 1987.
9. R. Johansen, User approaches to computer-supported teams, in M. H. Olson (ed.), *Technological Support for Work Group Collaboration,* Hillsdale, NJ: Lawrence Erlbaum Associates, 1989, pp. 1–32.

10. T. W. Malone et al., The information lens: An intelligent system for information sharing and coordination, in M. H. Olson (ed.), *Technological Support for Work Group Collaboration,* Hillsdale, NJ: Lawrence Erlbaum, 1989, pp. 65–88.
11. N. S. Borenstein, Computational mail as network infrastructure for computer-supported cooperative work, *Proc. CSCW'92,* New York: ACM Press, 1992, pp. 67–74.
12. Olivetti, The Pandora multimedia system, CSCW'92 Technical Video Program, ACM SIGGRAPH Video Series, Issue #87, 1992.
13. C. Perey, Desktop videoconferencing and collaboration systems, *Virtual Workgroups,* **1** (1): 21–31, 1996.
14. A. J. Sellen, Speech patterns in video-mediated conversations, *Proc. CHI'92,* New York: ACM Press, 1992, pp. 49–59.
15. K. Okada et al., Multiparty videoconferencing at virtual social distance: MAJIC design, *Proc. CSCW'94,* New York: ACM Press, 1994, pp. 279–291.
16. K. Okada et al., Design and evaluation of MAJIC videoconferencing system, *Proc. INTERACT'95,* London: Chapman & Hall, 1995, pp. 289–294.
17. Y. Ichikawa et al., MAJIC videoconferencing system: Experiments, evaluation and improvement, *Proc. 4th European Conf. on Computer-Supported Cooperative Work,* Dordrecht, The Netherlands: Kluwer, 1995, pp. 279–292.
18. T. Inoue et al., Learning from TV programs: Application of TV presentation to a videoconferencing system, *Proc. UIST'95,* New York: ACM Press, 1995, pp. 147–154.
19. S. Whittaker, Video as a technology for interpersonal communications: A new perspective, *SPIE,* **2417**: 294–304, 1995.
20. C. Heath, P. Luff, and A. Sellen, Reconsidering the virtual workplace: Flexible support for collaborative activity, *Proc. 4th European Conf. Computer-Supported Cooperative Work,* Dordrecht, The Netherlands: Kluwer, 1995, pp. 83–99.
21. G. Williams, Task conflict and language differences: Opportunities for videoconferencing? *Proc. 5th European Conf. Computer-Supported Cooperative Work,* Dordrecht, The Netherlands: Kluwer, 1997.
22. M. R. Macedonia and D. P. Brutzman, MBONE provides audio and video across the Internet, *IEEE Comput.,* **25** (4): 30–36, 1994.
23. E. A. Isaacs, T. Morris, and T. K. Rodriguez, A forum for supporting interactive presentations to distributed audiences, *Proc. CSCW'94,* New York: ACM Press, 1994, pp. 405–416.
24. E. A. Isaacs et al., A comparison of face-to-face and distributed presentations, *Proc. CHI'95,* New York: ACM Press, 1995, pp. 354–361.
25. T. Brinck and L. M. Gomez, A collaborative medium for the support of conversational props, *Proc. CSCW'92,* New York: ACM Press, 1992, pp. 171–178.
26. H. Ishii, M. Kobayashi, and K. Arita, Iterative design of seamless collaboration media, *Commun. ACM,* **37** (8): 83–97, 1994.
27. S. Greenberg, S. Hayne, and R. Rada, Groupware for Real-Time Drawing: A Designer's Guide, London: McGraw-Hill, 1995.
28. G. W. Dickson et al., Observations on GDSS interaction: Chauffeured, facilitated and user-driven systems, *Proc. 22nd Annu. Hawaii Int. Conf. System Sciences,* 1989, pp. 337–343.
29. J. F. Nunamaker, Jr. et al., Electronic meeting systems to support group work, *Commun. ACM,* **34** (7): 50–61, 1991.
30. C. J. McGoff and L. Ambrose, Empirical information from the field: A practitioner's view of using GDSS in business, *Proc. 24th Annu. Hawaii Int. Conf. System Sciences,* 1991, pp. 805–811.
31. N. A. Streitz et al., DOLPHIN: Integrated meeting support across local and remote desktop environments and LiveBoards, *Proc. CSCW'94,* New York: ACM Press, 1994, pp. 345–358.
32. G. Mark, J. M. Haake, and N. Streitz, The use of hypermedia in group problem solving: An evaluation of the DOLPHIN electronic meeting room environment, *Proc. 4th European Conf. on Computer-Supported Cooperative Work,* Dordrecht, The Netherlands: Kluwer, 1995, pp. 197–213.
33. P. Curtis and D. A. Nichols, MUDs grow up: Social virtual reality in the real world, *Proc. 1994 IEEE Computer Conference,* Los Alamitos, CA: IEEE Press, 1994, pp. 193–200.
34. D. A. Nichols et al., High-latency, low-bandwidth windowing in the Jupiter collaboration system, *Proc. UIST'95,* New York: ACM Press, 1995.
35. P. Curtis et al., The Jupiter audio/video architecture: Secure multimedia in network places, *Proc. ACM Multimedia'95,* New York: ACM Press, 1995, pp. 79–90. Also available online via ftp://ftp.parc.xerox.com/pub/MOO/papers/JupiterAV.ps
36. B. Damer, C. Kekenes, and T. Hoffman, Inhabited digital spaces, *Conf. Companion of CHI'96,* New York: ACM Press, 1996, pp. 9–10.
37. C. Greenhalgh and S. Benford, Virtual reality tele-conferencing: Implementation and experience, *Proc. 4th European Conf. Computer-Supported Cooperative Work,* Dordrecht, The Netherlands: Kluwer, 1995, pp. 165–180.
38. J. Bowers, J. Pycock, and J. O'Brien, Talk and embodiment in collaborative virtual environments, *Proc. CHI'96,* New York: ACM Press, 1996, pp. 58–65.
39. K. Andrews, F. Kappe, and H. Maurer, Hyper-G and Harmony: Towards the next generation of networked information technology, *Proc. CHI'95 Conf. Companion,* New York: ACM Press, 1995, pp. 33–34.
40. H. Maurer, *HyperWave: The Next Generation Web Solution,* Harlow, UK: Addison-Wesley, 1996.
41. T. W. Malone et al., Semistructured messages are surprisingly useful for computer-supported coordination, *ACM Trans. Off. Inf. Syst.,* **5**: 115–131, 1987.
42. T. W. Malone, K. Lai, and C. Fry, Experiments with Oval: A radically tailorable tool for cooperative work, *Proc. CSCW'92,* New York: ACM Press, 1992, pp. 289–297.
43. J. Conklin and M. L. Begeman, gIBIS: A hypertext tool for exploratory policy discussion, *Proc. CSCW'88,* New York: ACM Press, 1988, pp. 140–152.
44. J. M. Haake and B. Wilson, Supporting collaborative writing of hyperdocuments in SEPIA, *Proc. CSCW'92,* New York: ACM Press, 1992, pp. 138–146.
45. J. Grudin, Why CSCW applications fail: Problems in the design and evaluation of organizational interfaces, *Proc. CSCW'88,* 85–93, New York: ACM Press, 1988. Republished in D. Marca and G. Bock (eds.), *Groupware: Software for Computer-Supported Cooperative Work,* Los Alamitos, CA: IEEE Press, 1992, pp. 552–560.
46. J. Grudin and L. Palen, Why groupware succeeds: Discretion or mandate? *Proc. 4th European Conf. Computer-Supported Cooperative Work,* Dordrecht, The Netherlands: Kluwer, 1995, pp. 263–278.
47. R. Medina-Mora et al., The Action Workflow approach to workflow management technology, *Proc. CSCW'92,* New York: ACM Press, 1992, pp. 281–288.
48. C. A. Ellis, Information control nets: A mathematical model of office information flow, *Proc. ACM Conf. Simulation, Modeling and Measurement of Computer Systems,* New York: ACM Press, 1979, pp. 225–240.
49. P. Dourish, Developing a reflective model of collaborative systems, *ACM Trans. Comput.-Human Interaction,* **2** (1): 40–63, 1995.
50. P. Dourish, The parting of the ways: Divergence, data management and collaborative work, *Proc. 4th European Conf. Computer-Supported Cooperative Work,* Dordrecht, The Netherlands: Kluwer, 1995, pp. 215–230.

51. M. L. Markus and T. Connolly, Why CSCW applications fail: Problems in the adoption of interdependent work tools, *Proc. CSCW'90,* New York: ACM Press, 1990, pp. 371–380.

52. L. Suchman, Office procedures as practical action: Models of work and system design, *ACM Trans. Off. Inf. Syst.,* **1**: 320–328, 1983.

53. J. Bowers, G. Button, and W. Sharrock, Workflow from within and without: Technology and cooperative work on the print industry shopfloor, *Proc. 4th European Conf. Computer-Supported Cooperative Work,* Dordrecht, The Netherlands: Kluwer, 1995, pp. 51–66.

54. J. D. Gould, How to design usable systems, in R. M. Baecker et al. (eds.), *Readings in Human-Computer Interaction: Toward the Year 2000,* San Mateo, CA: Morgan Kaufmann, 1995.

55. J. Grudin, Interactive systems: Bridging the gaps between developers and users, *IEEE Comput.,* **24** (4): 59–69, 1991. Republished in R. M. Baecker et al. (eds.), *Readings in Human-Computer Interaction: Toward the Year 2000,* San Mateo, CA: Morgan Kaufmann, 1995.

56. S. E. Poltrock and J. Grudin, Organizational obstacles to interface design and development: Two participant observer studies, *ACM Trans. Computer-Human Interaction,* **1** (1): 52–80, 1994.

57. G. Bjerknes, P. Ehn, and M. Kyng (eds.), *Computers and Democracy—A Scandinavian Challenge,* Aldershot, UK: Gower, 1987.

58. J. Greenbaum and M. Kyng (eds.), *Design at Work: Cooperative Design of Computer Systems,* Hillsdale, NJ: Lawrence Erlbaum Associates, 1991.

59. D. Schuler and A. Namioka (eds.), *Participatory Design: Principles and Practices,* Hillsdale, NJ: Lawrence Erlbaum Associates, 1993.

60. H. Beyer and K. Holtzblatt, *Contextual Design: Defining Customer-Centered Systems,* San Francisco: Morgan Kaufmann Publishers, 1997.

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

62. W. J. Orlikowski, Learning from Notes: Organizational issues in groupware implementation, *Proc. CSCW'92,* New York: ACM Press, 1992, pp. 362–369.

63. C. Perin, Electronic social fields in bureaucracies, *Commun. ACM,* **34** (12): 74–82, 1991.

JONATHAN GRUDIN
University of California, Irvine
STEVEN E. POLTROCK
The Boeing Company

GUI.
See GRAPHICAL USER INTERFACES.

GUIDANCE OF MISSILES.
See MISSILE CONTROL; MISSILE GUIDANCE.

GUIDED-WAVE ELECTRO-OPTICAL SWITCHES.
See ELECTRO-OPTICAL SWITCHES.