Computer-Supported Collaborative Production

Jonathan Howarth[1], Beth Yost[1], Mary Beth Rosson[2], Philip Isenhour[1], John M. Carroll[2]
[1] Center for Human-Computer Interaction, Virginia Tech, Blacksburg, VA, USA
[2] Center for Human-Computer Interaction, Pennsylvania State University, PA, USA

Abstract

This paper proposes the concept of collaborative production as a focus of concern within the general area of collaborative work. We position the concept with respect to McGrath's framework for small group dynamics and the more familiar collaboration processes of awareness, coordination, and communication (McGrath 1991). After reviewing research issues and computer-based support for these interacting aspects of collaboration, we turn to a discussion of implications for how to design improved support for collaborative production. We illustrate both the challenges of collaborative production and our design implications with a collaborative map-updating scenario drawn from the work domain of geographical information systems.

Keywords

collaboration, collaborative production

Table of Contents

1 Introduction

When people report that they are collaborating, what do they mean? In practice, c*ollaboration* refers to a broad range of shared activities—chatting and discussing work, coordinating one's efforts with others working on the same project, even simultaneously editing a shared file. Because the term is overloaded with meaning, its use among different stakeholders can be problematic. Researchers working in the area of computer supported cooperative work (CSCW) are likely to understand a specific technical meaning for the term, but practitioners who purchase and use collaborative software work with a more fuzzy and informal understanding. The lack of a common vocabulary regarding what is meant by collaboration makes it difficult for these groups to communicate and determine wants and needs. As a step toward clarifying the broad concept of collaboration, we offer the term *collaborative production* to refer to group work that is directed at a final goal of *creating a shared product*. We survey current research and systems related to collaborative production and present a future vision of computer-based support for collaborative production.

1.1 Why Collaborative Production?

The literature and technology review and analysis reported in this paper arose from a joint project with Bearing Point, a business systems integrator, and Rosettex Technology and Ventures Group; we were working together on a project for the National Imagery and Mapping Agency (NIMA). The goal of the project was to evaluate the state of collaborative technologies used by NIMA. Projects at the agency often include shared work among professionals with differing expertise (e.g., geographical science, image processing, data mining) but integrated by a focus on a central shared artifact (e.g., an image map annotated with symbolic information). Our project deliverable was a "roadmap" for the integration of new research ideas and technologies into the agency. A key element of developing these recommendations was to first determine what aspects of collaboration were not adequately supported by existing technology.

In general, we observed that the commercial off-the-shelf (COTS) tools in common use, such as NetMeeting (NetMeeting) and IWS (InfoWorkSpace), limited workers to sharing, reviewing, and in some cases collaboratively editing files. These tools supported certain aspects of synchronous collaboration (e.g., working together on shared objects at the same time), but switching among tools for different tasks did not always facilitate, and in some cases even hindered, the final goal of many NIMA collaborative tasks: the creation of a useful product. This understanding helped us to distinguish the general communication and coordination activities of collaboration from a specific aspect of collaboration that is oriented toward the goal of *production*; we began using the term collaborative production to emphasize the concreteness of the goals driving this product-oriented style of collaboration. The collaboration in collaborative production is an integral part of the production effort, not an activity on its own.

Other researchers have critically examined the concept of collaboration and how collaborative tasks might be supported through technology of various sorts. For instance, Ellis and his colleagues offered an early framework that articulated the two basic dimensions of time (synchronous or asynchronous) and location (local or remote) (Ellis, Gibbs, and Rein 1991). Other analyses have highlighted task characteristics (Smith, Wolczko, and Ungar 1997), technology affordances (Benford, Brown, Reynard, and Greenhalgh 1996), and theoretical analyses of cooperative behavior (Malone and Crowston 1994).

Our work on collaborative production integrates ideas from these earlier frameworks. We have analyzed collaborative software with respect to whether and how they support a production-centered view of collaboration. This has led to a new view of the design space for collaborative production as well as ideas for new designs. In this paper we summarize our review and its implications in several sections. In the next section, we define collaborative production and describe in detail its components and dependencies. Following this, we survey existing research in several related areas of computer-supported collaboration, including a brief review of representative COTS technologies. After the survey, we discuss ideas and issues associated with tools or other support for collaborative production; we illustrate our discussion with a futuristic scenario in the domain of geographical information systems (GIS). Concluding thoughts are offered in Section 5.

2 Contribution

This paper contributes to IS research in three major ways. First, it summarizes and relates research on computer-supported collaboration. Second, it proposes and defines the idea of collaborative production, which to the authors' knowledge is a new concept in the field of IS research. Third, it discusses design implication and proposes a roadmap for tools that could better enable collaborative production.

The review of research describes a large number of research projects and several commercial off the shelf systems; it will be of interest to IS researchers working in areas that can benefit from collaboration support and practitioners who are considering investing in collaborative technologies or integrating them into existing systems or processes. The concept of collaborative production and the roadmap are expected to be very interesting to IS researchers. The concept of collaborative production distinguishes the general communication and coordination activities of collaboration from a specific aspect of collaboration that is oriented toward the goal of production while the roadmap highlights research areas that need to be explored in the new design space created by the concept of collaborative production.

3 Collaborative Production

In the computer science literature, researchers often simply assume an understanding of what it means to collaborate, as illustrated by Suchman and Trigg (1986): “The term ‘collaboration,’ like many of the words used to classify everyday activities, is an abstraction over a rich array of concrete practices.” Because it encompasses so much, collaboration is difficult to define, particularly in terms of what it is *not*. With our concept of collaborative production, we do not intend a comprehensive definition of collaboration, but instead a scoping of collaborative activities that focus on creation of shared products. For the purposes of this paper, we consider a *product* to be an artifact that is created for a purpose and that can be made accessible to others.

Our interest in shared production builds on the seminal research on teamwork by Joseph McGrath and his colleagues (1991). These researchers described a range of *functions* (production, group well-being, member support) and *modes* (inception, problem-solving, conflict resolution, execution) that characterize the behavior of groups at different points in their collaboration. More recently this framework has been elaborated to also consider team formation, maintenance, and evolution (Arrow, McGrath, and Berdahl 2000). We have illustrated the basic framework in Table 1, instantiating McGrath’s function-mode matrix with a hypothetical GIS (geographical information systems) scenario drawn from our work with NIMA—a map-updating activity initiated by a new treaty that mandates adjustments in national borders.

Table 1: Modes and functions of TIP Theory for a hypothetical GIS scenario

	Production	Group Well-being	Member Support
Inception	Ed, a NIMA manager distributes a memo outlining implications of a new treaty	Two treaty analysts (Anna, trainee Bill) and two map experts (Carla, Don) are asked to coordinate the update	Prospective members check their schedules; analyze their likely contributions and Ed's expectations
Problem-solving	Team discusses requirements, outlines subtasks, possible schedule, etc.	Ed and Anna suggest Bill take the lead in translation; Carla and Don will share map updating	Members reflect on how they are making contributions, what rewards will result in short and long term
Conflict Resolution	Team identifies and negotiates priorities and dependencies in their subtasks	Ed offers leadership while also accepting others' refinements or alternatives	Members propose and argue for personal or subgroup priorities and resources
Execution	*Bill drafts translation; Anna checks and refines, hands off to Carla, who annotates old map; Don enters changes; as a group they review/confirm*	Anna gives feedback to Bill on his translation; Carla and Don go over her annotations; Ed checks in with each team member as the tasks progress	Bill learns some new terms; Don reuses a notation system he has used earlier; Ed is commended for his team's quick response

Perhaps the most salient function of any team collaboration is *production*: the ultimate goal of this hypothetical team is to generate the revised map. But the actual implementation of the revisions takes place only in the final phase of the team's efforts; prior to that they reach a shared understanding of their mission, develop a technical plan for achieving the mission, and identify and address conflicts in the implementation strategy to be followed. In McGrath's (1991) framework, these phases are implicit in achieving any production goal, even though the preliminary phases may be short-circuited when the project is familiar, the team has achieved similar goals in the past, and so on.

While the focus of collaborative work is often on production, an important contribution of McGrath's framework is that it highlights the role of group well-being and member support in the successful activity of a group. Group well-being efforts focus on ensuring the coherence and sustainability of the team as a configuration of collaborators; just as important are the group's efforts toward developing and maintaining the abilities and motivation of its individual members.

The scenario illustrated in Table 1 also helps to focus our definition of collaborative production, namely on the *execution mode of the production function*, where team members are carrying out behaviors—on their own or together—that instantiate and shape their final product. Drafting and revising the treaty translation; annotating an older artifact; entering changes to create a new artifact; reviewing and accepting these changes: all of these are subtasks in the collaborative production of the new and improved map.

At the same time, the TIP framework emphasizes the aspects of teamwork that are not part of collaborative production, but that are critical in successful execution of a shared task. Developing a shared understanding, assigning and negotiating roles and individual contributions, determining and administering rewards or other feedback as subtasks are completed: these are essential elements of smooth group functioning (Schmidt and Simone 1996).

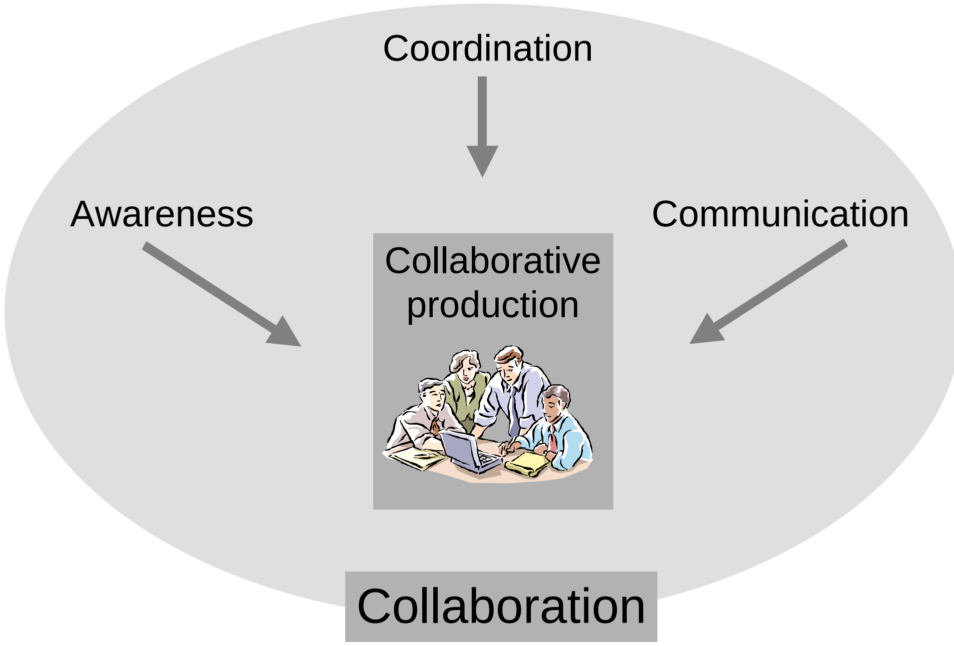

Figure 1: Aspects of shared activities that enable collaborative production

In the remainder of this section we discuss these dependencies in terms of three basic requirements for shared activity (Figure 1). Potential or already-engaged collaborators must acquire and maintain *awareness* of one another's goals, plans, work status, communication availability, and so on (Carroll, Neale, Isenhour, Rosson, and McCrickard 2003). Assuming that they possess a basic level of awareness, collaborators can *coordinate* their efforts, expending effort on the "articulation work" needed to negotiate shared work, schedule and re-schedule tasks, hand-off intermediate products, and monitor (Schmidt and Bannon 1992). In support of both awareness and coordination, collaborators identify and use appropriate *communication* mechanisms (Olson and Olson 2000). The work thus carried out as a group can be broadly defined as *collaboration*, with collaborative production seen as the product-focused components of collaborative activities. As suggested by the figure, even though shared creation of a product may be the ultimate goal of a group, success in achieving this goal will depend on adequate support for awareness, coordination, and communication (see McDonald, Weng, and Gennari 2004 for a documented example of the need for this support). We turn now to a brief introduction to these three aspects of shared activity and their impacts on collaborative production. We illustrate these relations using a familiar collaborative production scenario, that of co-authoring a research paper.

3.1 Awareness of Others and the Situation

A precursor to active collaborative work is awareness—both *social* awareness (Who is part of the team? Who is here now?) and *activity* awareness (What has happened so far? What is happening now?). Before people can engage in meaningful, directed exchanges, they need to be aware of one another. Social awareness is an understanding of who is available for interaction and what they know. Activity awareness, as the term implies, is an understanding of the activities of others and how those activities relate to a desired task. While social awareness provides a contact network, activity awareness provides a context for action (Dourish and Bellotti 1992). The level of social and activity awareness may vary based on the task, but some minimal level of each is necessary.

In the task of co-authoring a research paper, awareness serves an important introductory role as well as a maintenance role. Initially, a researcher must become aware of existing research to discover other researchers who are working in similar areas and to identify possible contacts or collaborators. A major part of this process is determining which researchers are already actively constructing or testing research systems and how developed these systems are. After the commencement of the research activity, researchers must remain aware of the state of his or her own research, the paper that is being developed to document the research, as well as the contributions and further intentions of his or her collaborators.

3.2 Coordinating Work with Others

Coordination is required for any activity that involves multiple inputs from multiple individuals (Schmidt *et al.* 1996). We use the definition offered by Malone and Crowston (1994): "Coordination is managing dependencies between activities." Assuming that a team's members have identified one another as collaborators and have agreed to work together, they can then focus their efforts on a collaborative task. This focusing process involves coordination; concurrent efforts that are not coordinated may be highly inefficient or even in conflict with one another.

With respect to collaborative production, coordination encompasses two main activities: scheduling interaction and managing coupling issues. Regardless of the technology used to collaboratively perform work, *ad hoc* or opportunistic interactions are rarely efficient methods for coordinating effort. Predetermined interactions, on the other hand, provide group members with an opportunity to create an agenda and prepare themselves and their materials. An extension of the idea of scheduling interactions is synchronizing effort to handle coupling issues. Traditional devices such as Gantt and PERT charts allow group members to divide their work, so that they can parallelize their efforts. When group members are aware of dependencies, they can most efficiently divide their time among required tasks.

In the collaborative research example, if communicating researchers decide that their work is sufficiently similar, they may commit to integrating their efforts. Coordination allows the researchers to schedule future communications and develop a plan for sharing their intermediate results as well as integrating their final results.

3.3 Communicating with Others

An aspect of collaboration that supports all others is communication. The nature of the communication process, the task functions of a group, and the actual properties of the medium chosen for communication affect the quality of the communication (Daft and Lengel 1986; McGrath 1991; Dennis and Valacich 1999). Being aware of a collaborator and his or her activities may motivate a person to initiate communication, choosing the medium that best matches the type of information to be communicated as well as the potential needs of the group.

With respect to computer support for collaborative production, intra-team communication is likely to rely at least partially on computer-mediated communication (CMC). Research on CMC focuses not only on the technologies that enable communication, but also on the sociality of work. By enabling communication, technologies such as discussion boards, email, instant messaging, and audio and video teleconferencing create changes in the social fabric. They may create new opportunities by attenuating imbalances in power and initiative, but they also result in potentially harmful interdependencies and introduce misunderstanding (Sproul and Kiesler 1991; Olson *et al.* 2000).

Again considering our collaborative research and authoring example, communication needs and opportunities are promoted by awareness, but at the same time awareness is enhanced through effective communication. For instance after becoming aware of a potential collaborator, a researcher may communicate to him or her via email or in person at a research meeting. As the project develops, communication will also be important for allowing group members to share their descriptions and interpretations of research methods and results. Such communication allows researchers working as a group to maintain a consistent understanding of their objectives and make progress towards those objectives.

3.4 Collaborating to Create a Product

We have distinguished between collaboration in general and collaboration to create a product. Group members often work together without a concrete production goal. For example, not all research paper review or analysis have a production function; on occasion, group members may complement a peer's work, contributing to the group's member support function, or they may remind one member of his or her job as the statistics expert, contributing to group well-being (McGrath 1991; see also Table 1). Collaboration involves working together, but not always *doing the work* together.

Collaborative production may be co-located or remote; the work may be carried out synchronously or asynchronously. Group members may operate on a single instance of an artifact or they may manipulate separate instances using a range of different tools to exchange results and maintain consistency.

In the next four sections, we survey survey research issues, systems, and technologies related to computer-based support of awareness, communication, coordination, and collaborative production. When surveying existing systems, we recognize that not all types of systems fit neatly into one of the categories. In cases where multiple categorizations are possible, for example a research system that addresses issues of both communication and awareness, we discuss it in the category in which it seems to have the strongest implications for collaboration. We focus on recent research, primarily work done after 1999, because earlier collaborative systems are well documented and their ideas have been integrated into the more recent work we report. In a few cases we include seminal papers that provide important background for current work. After reviewing this research literature, we briefly discuss COTS systems that support collaborative work, then turn to a discussion of design implications for collaborative production.

4 Computer-Mediated Collaborative Awareness

We define awareness as the perception of *meta-data* about collaborators that is external to task execution. The technologies that convey such data to collaborators fall into two main categories, 1) awareness displays and 2) filtering systems. *Awareness displays* are representations intended to make collaborators aware of what their partners are doing. We present current research on awareness widgets used on the desktop, those that go beyond the desktop, including findings from empirical studies. In contrast, *filtering systems* direct content to collaborators based on criteria such as user preferences or group activity histories. In this category, we also consider recommender systems that help users locate content and make decisions involving the content. A pervasive issue in current research on awareness is privacy.

4.1 Awareness Displays

Many awareness visualizations and information displays are intended for use in a desktop environment. For software developers, COLLIDE uses various visualizations to allow programmers to view who is working on what source code (Penner and Gutwin 2003). A specific and increasingly common example is *radar overviews*, which show where each user is working with respect to the entire information space (Gutwin and Greenberg 1999). This functionality is available through GROUPWEB, a tool that allows users to see where other users are working in a large two-dimensional workspace. GROUPWEB includes a collaborative web browser with awareness features built into the scrollbar for displaying remote collaborators' viewports; it also provides the option to synchronize with other users' views (Greenberg 1997). The scrollbar awareness techniques that are used are applicable to other workspace-oriented applications. GROUPWEB also includes a shared editor for annotations.

Other research efforts on awareness displays for the desktop provide a more personal approach. Awareness information can be provided using pictures of collaborators or their faces. For example GROUPSPACE allows multiple users to interactively view images of their collaborators; each user is represented by a 3D model of a head that is adjusted as his or her view changes (Dyck and Gutwin 2002). In the INSTANT MESSENGER VISUALIZATION, pictures of remote collaborators are placed in a visualization in which they move closer to the user as they become more available; this proximity awareness helps users monitor remote collaborators and know when to initiate communication (Neustaedter 2001).

Some awareness visualizations are intended for use beyond typical desktop displays. The TOWER system provides awareness of activities and context through symbolic presentations in a 3D environment (Prinz *et al.* 2002). Greenberg and Rounding (2000) and describe real-time collaborative surface techniques designed for large screen displays. The NOTIFICATION

COLLAGE—a system that developed for internal use by a collaboration awareness research group—allows users to share photos and other images of coworkers and friends. It was designed to promote informal and opportunistic communication and coordination as well more task-focused activities. Physical surrogates are another approach to awareness widgets (Greenberg and Kuzuoka 2000); they provide tangible representations of remote partners. These interfaces provide awareness information that is easy to understand, support the transition into conversation, balance the amount of information, and consider privacy issues.

A range of empirical studies have examined the impacts of awareness information on collaboration. In one study, four methods of representing changes in UML diagrams were tested, including replay changes, a chronological storyboard, iconic changes, and textual documentation. Most participants preferred the documentation method (Tam, McCaffrey, Maurer, and Greenberg 2000). Another study observed the use of a groupware system and found that conventions such as naming files and who replaces who in cases of absence are facilitated by awareness information (Mark, Fuchs, and Sohlenkamp 1997). Boyle, Edwards, and Greenberg (2000) tested blurred and pixelized video filters to compare tradeoffs in awareness and privacy (i.e., the more aware is of one's partners and their activities, the more intrusion or privacy invasion they may experience). These researchers found that a blurred filter can provide awareness and still protect privacy and that the pixelized filter can as well, but to a lesser extent.

4.2 Filtering and Recommender Systems

Filtering and recommender systems are designed to guide users to information as well as protecting them from information overload. Wolverton (1999) describes the importance of filtering and retrieval tools in fast-changing collaborative environments and presents a Task-based Information Distribution Environment (TIDE). This tool can be used to deliver only the documents that are relevant to a user’s current tasks. However TIDE does not take into consideration the tasks of other users in the collaborative environment.

Breese, Heckerman, and Kadie (1998) present various filtering algorithms that recommend content or services to a user based on the analysis of the behavior of a large number of users. Similarly, Payton, Daily, and Martin (1999) describe tools used to identify other web users with related interests based on both long-term slow decay and short-term fast decay. Privacy and disclosure, however, remain major issues. Another issue is robustness or how well collaborative filtering algorithms perform in the presence of noise. O'Mahony, Hurley, Kushmerik, and Silvestre (2004) discuss the stability and accuracy of a widely used filtering algorithm. CYCLADES allows scholars to search heterogeneous, distributed archives for information and also includes filtering and recommender technologies (Gross 2003). Another approach is to use physical proximity. For example, Ferscha, Holzmann, and Oppl (2004) discuss a subsystem that maintains location information on users of mobile computers and a corresponding instant messenger service that leverages the subsystem's information to facilitate the formation of groups based on physical location.

Notification systems are a subset of filtering and recommender systems that use user- or system-defined rules to filter the information a user receives (McCrickard 2003). For example, Horvitz, Jacobs, and Hovel (1999) use attention-sensing alerting to determine what notifications to send to users based on the current work of the user. They use Bayesian models to infer a probability distribution over a user's focus of attention and infer the potential cost of transmitting alerts to users. Note that while we point to notification systems as a technique for supporting awareness, we do not include a comprehensive survey because such systems are themselves a large and evolving research topic (McCrickard 2003).

5 Computer-Mediated Communication

Computer-mediated communication (CMC) has an extensive research history in both tool development and empirical study. The range of tools for CMC is broad, but basic technologies include discussion boards, email, instant messaging (IM), and audio and video teleconferencing. Here we focus on tools that have been designed to facilitate communication in support of collaborative activities.

Some CMC technologies focus on facilitating communication by creating one or more channels for communication in a collaborative setting. For instance, CORONA provides scalable group communication abilities for collaborative systems, building in support for different modes and roles in collaboration. The system has two modes for disseminating information: publish and subscribe, in which a recognized publisher sends content to multiple anonymous subscribers; and peer group, in which the peers are aware of one another (Hall, Mathur, Jahanian, Prakash, and Rassmussen 1996). The EUREKA system in use at Xerox allows service technicians to share tips not found in manuals (Roberts-Witt 2002). The tips are confirmed before being posted, and the employee's name is stored along with the tip. This technology has formed the basis for the LINKLITE knowledge-sharing system, a system that extends the tip-sharing concept to other work settings.

Other systems facilitate communication between heterogeneous devices. Tang *et al.* (2001) use awareness information to facilitate communication in CONNEXUS, a tool that integrates IM capabilities into desktop applications. A related tool, AWARENEX, extends the functionality of CONNEXUS to handhelds. Krebs, Ionescu, Dorohonceanu, and Marsic (2003) discuss the development of a collaborative system that handles access by heterogeneous clients. Data distribution agents select what data should be sent to users based on the users' access properties such as available bandwidth and reliability of the connection. Also, the shared objects are displayed based on the capability of the device. A study presented by Velez *et al.* (2004) discusses communication as it affects performance and interaction when collaborating participants use different platforms (either PC or PDA).

While the systems mentioned in the last two paragraphs make communication possible, other systems are aimed at making communication more efficient. For example Aoki *et al*. (2003) present a social "mobile audio space" that can be used to support multiple simultaneous conversations. The intent of this project is to allow mobile groups (e.g., five friends) to communicate using audio channels throughout the day (see also Ackerman, Hindus, Mainwaring, and Starr 1997). The challenges include privacy issues and potential irritation at hearing everyone's conversations all the time. One sub-project supports multiple simultaneous conversations by lowering the volume of users whose conversation analysis does not seem to fit a typical turn-taking pattern. A different project studies how people use two-way, push-to-talk cellular radios.

Another system, KANSAS, provides communication abilities based on proximity (Smith *et al.* 1997). Users form different audio and video connections based on their relative proximity to one another in a 2D world. Bradner, Kellogg, and Erickson (1999) examine and discuss how collaborators' technical configurations enable different types of communications based on a field study of their collaborative tool BABBLE. Another tool aids in remote videoconferencing by providing users with the sense that they are actively interacting and walking around in a teleconferencing room with one another (Perry and Agarwal 2000).

6 Computer Support for Coordination

Systems for coordination fit in one of two basic categories: technologies that allow for human resource management and technologies that allow for document management. Human resource management technologies allow group members to plan meetings and divide work. Examples include shared calendar systems and project management systems like GROUPPLAN, a multi-user, distributed, project planning environment (Blum and Gutwin 2003). Because the goal of collaborative production is the creation of artifacts, we focus our review of coordination tools on those concerned with document management.

Document management systems allow group members to plan and coordinate their contributions in terms of product creation. They are the predominant technology for managing shared products, allowing users to work asynchronously and then integrate their work. Rees, Ferguson, and Virdhagriswaran (1999) discuss the major issues involved with document management; they focus on erasing and overwriting, two main causes of information loss that may change the context of another user's work. Their proposal is that each change should be accompanied by a structured description and consistency conditions that must be met for the change to make sense.

A common scenario for document management is a collaboration that is *ad hoc* but must take place in a secure and reliable fashion. In these cases, one approach is to enable file sharing by allowing documents to reside on a single person's machine and to distribute them to collaborators only when and as necessary (Berket and Agarwal 2003). The master documents are updated automatically if collaborators make changes. Ionescu, Krebs, and Marsic (2002) describe a specific algorithm that allows users to work offline with a shared document offline, but then to update the master copy of the document when they reconnect. ICECUBE is another document management tool that uses a log-based approach to reconcile divergence in separate instances of a document (Kermarrec, Rowstron, Shapiro, and Druschel 2001). Shen and Sun (2002) describe their FORCE prototype, a tool for asynchronous collaboration that distinguishes semantic document merging policies from syntactic mechanisms.

7 Computer-Mediated Collaborative Production

A number of collaborative systems have been built in recent years with the goal of helping group members carry out their production activities together. In this section, we first survey toolkits and architectures that enable construction of such systems. We then discuss two approaches to collaborative systems—collaboration aware and collaboration transparent systems. Next, we briefly survey recent work on collaboration between heterogeneous devices and undo/redo issues in collaborative work (see also Mills 1999).

7.1 Architectures and Toolkits

Dewan (1999) surveys collaborative architectures and issues associated with them. Laurillau and Nigay (2002) discuss the CLOVER architecture that defines production, communication, and coordination services; they illustrate the architecture's usefulness with the COVITESSE system. The EFT framework combines ideas from the Collaborative Objects Coordination Architecture (COCA) and CAB, the Collaborative Active WhiteBoard (Li, D. and Patrao 2001). Chung and Dewan (2004) have analyzed and prototyped dynamic architectures that automatically switch among centralized, replicated, and hybrid architectures based on resource availability.

Toolkits aid in the development of collaborative applications by providing designers with basic functionality for common tasks. Roseman and Greenberg created GROUPKIT (1996), a multi-platform toolkit in the TCL/TK language for building real-time collaborative applications. It includes collaboration-aware components such as multi-user scrollbars along with session management and basic support for data replication. Shared graphics primitives are designed to simplify construction of shared 2D workspaces. GROUPKIT is no longer under active development, but it has influenced many other research projects. A similar tookit is DISTVIEW, which provides for the sharing of views at the window level (Prakash and Shim 1994). Multiple window applications can choose which windows to share and which to keep private.

7.2 Collaboration Awareness

Systems that support collaboration can do it in one of two ways: *Collaboration aware* systems are built explicitly to support collaborative activities; they may be a single application or an integrated environment containing multiple applications. In contrast, *collaboration transparency* systems enable collaboration that takes place with single-user applications, without modifying the applications. They often share the data and user input of a single user application through screen scraping techniques or build a shared multi-user data object that is external to the application itself.

A recent example of a collaboration aware system is GROUPMORPH, a collaborative editing system that allows the formation of groups and subsequent shared manipulation of 3D objects (Linebarger and Kessler 2002). Another example is TEAMROOMS, a persistent workspace-oriented collaborative environment that supports a variety of authoring and communication tools (Roseman and Greenberg 1997). (This tool has been commercialized as TEAMWAVE and later bought by Sonexis).

Billinghurst and Kato (1999) discuss the concept of collaborative mixed reality. *Mixed reality* is an interleaving of virtual objects and objects in the real world. The motivation to create a mixed reality system for collaboration comes from understanding the potential discontinuities among alternate workspaces, as well as the differences between new and existing work processes. Mixed reality has potential for both remote and co-located collaboration activities.

The groupware architecture used to build collaboration aware systems is an important factor in designing and implementing new tools. A framework and the applications that adhere to it often have to adapt to changing requirements; for this reason Yang, Inbarajan, and Li (2003) advocate a flexible component-based approach in which collaborative systems are constructed out of loosely-coupled components that are collaboration aware, but that do not know about one another. Such a framework could allow COTS components to be used in a plug-and-play fashion.

Collaboration transparency occurs when a single-user application is used for collaborative work without changing the original source code. Collaboration transparency has high practical significance because it enables collaborators to continue working with the applications with which they are already familiar. For example, Begogle, Rosson, and Shaffer (1999) describe FLEXIBLE JAMM, a system that replaces single user components with multi-user components at runtime. A limitation of the FLEXIBLE JAMM approach is that it only works for single-user applications that use a well-defined Java component architecture. In more recent efforts, Xia, Sun, Sun, Chen, and Shen (2004) discuss work on COWORD, a system that allows users to simultaneously edit one Microsoft Word document by managing and propagating updates to a central document object. In contrast to COWORD's approach of homogeneous application sharing, D. Li (2001) presents the idea of intelligent collaboration transparency (ICT), an ongoing project that covers heterogeneous application sharing and interoperation. ICT also addresses unconstrained interaction with application sharing systems, and incremental acquisition and formalization of application knowledge. Both CoWord and ICT use operational transformations to maintain consistency among heterogeneous applications.

7.3 Collaboration across Heterogenous Systems

Methods for sharing among heterogeneous applications have become increasingly important, as the number and diversity of end-user applications has grown. The general problem is discussed by Dewan and Sharma (1999), who analyze issues of concurrency, coupling, and architecture. As an example, these authors showed how a Habenero spreadsheet and a UNC spreadsheet could be used by different users to synchronously edit the same underlying spreadsheet object. Jackson and Grossman (1999) describe a system – also based on Habanero – that has support for heterogeneous client views and persistence via event reply. The survey-based persistence that supports asynchronous collaboration uses either state dumps or replay. Captured audio and video are also mapped to application events.

DeepView allows multiple users to share scientific instruments and collect, process, and store information from them (Parvin, Taylor, Cong, Okeefe, and Barcellos-Hoff 1999). Both synchronous and asynchronous collaboration are possible. The themes are functionality, scalability, and performance, and the architecture allows for COTS technology to be incorporated. A list of active users is provided, and users can share large amounts of data, chat, and exchange graphic overlays and images. It is also possible to transparently share heterogeneous single-user applications using this same approach of intelligent collaboration transparency (Li, D. and Li 2002).

7.4 Supporting Undo and Redo

When building support for collaborative production, the provision of undo and redo services becomes a significant concern. This of course is most noticeable during synchronous collaboration, when partners working together may enact conflicting actions at the same or near-same point in time. Choudhary and Dewan (1995) consider the general problem of multi-user undo/redo, and Perry, Agarwal, and McParland (2002) contribute scenarios in which collaborative editing might be used for scientific collaboration. C. Sun and Chen (2002) discuss creating multiple versions of graphics objects to handle the conflicts that result as part of concurrent editing. They offer GRACE as an Internet-based implementation. Chang (1998) describes the design and rationale for SPARROW, a system that allows multiple people to simultaneously edit the same website. SPARROW users are able to change information without knowing HTML because they do the editing within a browser.

The algorithm used to decide *what* to undo or redo is an important design concern. An undo algorithm presented by C. Sun (2002) is part of a much larger research project concerned with *operational transforms* (OT). D. Sun, Sun, and Chen (2004) present an extension to the earlier algorithm and show how it was applied in the development of COWORD. R. Li and Li (2003) also discuss an OT algorithm that allows collaborators over a network to simultaneously edit any part of their local replicas of documents without delay and with response as good as a single-user editor. The algorithms that this group is working on include SDT (state difference transformation), TIBOT (Time Interval Based Operational Transformation), and regional undo. The project also aims to make current OT algorithms more usable and useful. An especially challenging area for concurrent editing is in bitmap based graphics editing. Wang, Bu, and Chen (2002) provide a useful overview of the problems involved as well as offering an algorithm that should be able to undo any operation at any time.

8 COTS Tools for Collaborative Work

There has been a relatively slow cycle of incorporating research in collaborative production into useful COTS tools. In particular, commercial tools still reflect a view of collaboration as a separate activity, somehow independent of a group's task-oriented goals. The tools tend to be meeting-oriented and are usually intended for short, preplanned collaborative sessions. In these tools, extended sessions that include spontaneous or ad-hoc collaborations are impractical, because of the high degree of overhead in maintaining the collaboration context.

As an illustration of the state of COTS tools, we discuss four popular collaborative applications that span the range of what is currently available. Microsoft’s NETMEETING (NetMeeting) and Advanced Reality’s PRESENCE-AR (Presence-AR) are collaboration transparency systems, while Documentum’s eROOM (eRoom) and Groove Network’s GROOVE (Groove) are collaboration aware.

One of the earliest approaches, and one that is still used heavily, is the sharing of single-user applications through a broadcasting mechanism that sends graphical output from one application to other users’ displays. Microsoft NETMEETING is the most popular example of this technology. Although this approach allows group members to make use of familiar applications, it has certain key limitations including view synchronization requirements, input restrictions, and huge coordination costs. Group members in an application-sharing session are limited to exact view synchronization; they will see exactly the same display. This means that members cannot independently scroll to separate sections of a shared document. Accordingly, application-sharing is likely to only be useful for the most tightly coupled activities. A second limitation is that only one group member can be modifying the document at any given time (the underlying single-user applications assume a single stream of input). For example, the blinking text cursor in a word processor can only ever exist in a single place. While application-sharing tools handle floor control transitions, switching control of input from one user to another, in different ways and with varying degrees of transparency, the fundamental turn-taking constraint presents a significant usability barrier in all but the shortest and most focused collaborative editing tasks.

The third limitation concerns the coordination overhead that application-sharing requires. Because this technology is strictly synchronous, group members must coordinate their schedules to find a time to connect. Because the technology involves transmission of graphical information, application-sharing tools can be bandwidth-intensive, and their use may be constrained by network management policies. Collaborators must determine who should run the “live” copy of the application, and must ensure that the outcome of the session is distributed to participants or posted to shared file storage. Better session management tools and integration with document management systems can, at best, help reduce this overhead.

Other COTS approaches to collaboration transparency have attempted to address these issues. For example, PRESENCE-AR does not attempt to share the application itself, but instead shares only the application's data. Shared data objects are created and stored outside of the application. Each user runs his or her own instance of the application, and all applications update and receive updates from the shared data object. This approach does solve some of the problems of application-sharing, but introduces problems of its own. The most obvious of these is that shared data objects must be developed for any application used as part of a collaboration activity. These data objects are often proprietary, such that reverse engineering may be necessary to analyze and reconstitute the appropriate data. A more subtle problem is that collaboration is still treated as an add-on feature and not an integrated part of the use of an application.

eROOM and GROOVE take a different approach. These systems are environments designed from the beginning to include collaborative tools such as calendars, discussion boards, and text objects. eROOM is web-based and intended primarily for asynchronous communication, while GROOVE is a client application that supports some synchronous editing. Having an integrated environment alleviates the need to support already existing applications and ensures that all tools are collaboration aware. The main problem with these tools is that it may be unreasonable to expect users to give up their favorite or most familiar applications simply for the collaborative abilities offered by tools in the environment. Collaboration is once again a special feature that serves to disrupt user practices. Orlikowski (1992) discusses employee resistance to the adoption of LOTUS NOTES. Such resistance is likely to occur whenever users are forced to collaborate through the use of a tool that they would not otherwise choose.

9 Design Implications

Given the state of research and COTS technologies, how can we facilitate the integration of collaborative features as an inherent element of productivity tools? Only through integration will users experience collaborative production (and indeed collaboration more broadly) as less of a specialized activity and more of a standard practice. In the following sections, we offer a roadmap for tools that could better enable collaborative production. To simplify our discussion of design implications, our roadmap focuses on a specific form of collaborative production: concurrent editing. We discuss the role of task coupling in collaborative production work such as concurrent editing and issues for further research on concurrent editing systems and collaborative production in general. To illustrate the potential of our ideas, we return to the GIS work domain, considering implications for the map-updating scenario.

9.1 Task Coupling in Collaborative Production

Our survey of research and COTS systems points to the critical role of coupling in determining how best to support collaboration: collaborative tasks that are tightly coupled (e.g., proofreading) have different sharing requirements than those that are moderately or loosely coupled (e.g., editing different sections of the paper). We partition the concept of collaboration into the following three overlapping categories depending on the degree of coupling between collaborators: parallel production, dependent production, and interdependent production. We describe each category and then discuss implications of how fully-supported concurrent editing would address this style of co-production.

9.1.1 Parallel Production

Parallel production refers to concurrent modification of independent parts of a product. In parallel production, group members are typically working independently on distinct products or different parts of one product. The degree of coupling is determined by how the boundaries of a given group member's work area are managed. This approach to collaboration can be highly efficient, but there are two primary sources of inefficiency. The first is the merging of individual work once all the parallel efforts have concluded. The exact nature of this merging operation and extent to which it delays final output can vary based on the diversity of tools, data sources, and products handled by individual group members. The second source is the "edge problem." The boundaries of a user's work area may be fuzzy and group members may duplicate effort or produce incompatible products for the same area.

Technology for fully-concurrent editing would address these issues. All users would operate on the same data set, at the same time or different times, eliminating the need for post-hoc merging. The edge problem should also largely disappear because group members working on neighboring areas will be able to spot potential problems and decide, when appropriate, who should cross the boundary to produce the best representation of an edge area.

9.1.2 Dependent Production

In contrast to parallel production, dependent production refers to concurrent effort on parts of a product that depend on each other, such as critiquing another team member's written work. Ideally, by making work visible group members at later stages of a project can work more efficiently because they will be able to see what has happened at earlier stages. Conversely, members who worked on earlier stages may be able to more quickly spot misinterpretations of the source materials that they produced. Often efforts are made to parallelize work, but for dependent parts of a product, parallelization may be wasteful. One reason is that for many tasks, a high degree of synchronization may limit productivity. Minimally, the ability to view dependent products as they are being generated allows for more detailed planning of later-stage production activities. In some cases, it may even be practical to allow later-stage activities to be initiated with only partial completion of earlier stages. A second reason is that parallelization may isolate group members and eliminate opportunities for feedback during production.

Concurrent editing capabilities would give collaborators at different stages of the artifact's production some degree of visibility into upstream and downstream progress. With tighter coupling of activities, awareness becomes a greater consideration, allowing users to easily discover where current work is being done and where recent work was done.

9.1.3 Interdependent Production

A third style of co-production is interdependent production, in which group members manipulate the same area of the same product at the same time. As we noted in our survey of COTS tools, most commercial efforts directed at collaboration tools have focused on technology that supports distributed meetings and tightly coupled collaborative interaction within the same data. In addition to audio and video communication channels, most offerings focus on the use of application-sharing packages or integrated environments.

Usability issues and inherent coordination overhead make current approaches to application-sharing impractical for ongoing, *ad hoc* collaborative production activities. True concurrent editing approaches that allow relaxed view synchronization, distributed simultaneous input, and that do not require explicit coordination of sessions have the potential to support the kind of focused, meeting-oriented interactions currently done with application-sharing while allowing the evolution of more flexible collaborative practices.

9.2 Open Issues

Our analysis of concurrent editing issues suggests that we need to develop approaches to four concerns. First we need to support concurrent editing in a way that is automatic and visible only when direct collaboration is needed. Eliminating the tool-switching and disruption required to engage in collaboration as a separate process should lower the cost of accepting the technology and provide sufficient flexibility to allow new collaborative processes to evolve. Second, we must support focused interdependent production activities as a natural extension of day-to-day work, requiring minimal overhead and disruption. Third, we need to provide sufficient awareness and communication features to allow fluid, *ad hoc* transitions between loosely- and tightly-coupled activities. Finally, we should leverage established document management technologies for long-term data management and asynchronous collaboration. We discuss issues related to these goals both for the the enabling processes of awareness, coordination, and communication, and for collaborative production.

9.2.1 Awareness, Coordination, and Communication

If a group of distributed individuals is editing the same document at the same time, they may experience confusion and frustration, because changes made by remote group members—and the rationale for these changes—may not be apparent. Opportunities for collaboration may also be missed if members are unaware of their remote colleagues' presence, actions, and intentions. Commercial and research systems address various facets of these problems. Tools that support synchronous interaction often provide integrated text, audio, and/or video channels, which are useful for resolving questions about remote changes. Searching, filtering, and versioning tools in recommender systems and document management tools can provide awareness information that extends beyond synchronous interactions.

There are several open questions, however, concerning the integration of awareness, coordination, and communication capabilities. One question is whether workspace awareness techniques such as radar panes and telepointers are suitable for all kinds of data, particularly new types yet to be developed. Another question is whether existing recommender systems and document management technologies can take advantage of natural properties and associations of data to establish relationships or if new techniques for relating data need to be developed. Finally, privacy and security constraints are inherent in a group production environment. It is important to learn, given these constraints, what kinds of information can be integrated into social awareness tools to facilitate opportunistic interaction among distributed analysts.

9.2.2 Collaborative Production

The concurrent editing of documents or other data objects introduces a number of issues. On a lower level, it is important to enable concurrent editing for both existing data objects and data objects yet to be developed. Operational transforms are, at least in principle, capable of providing this service. On a higher level, it is desirable to make such collaborative services invisible, so that they can become a natural part of work. Invisibility, however, introduces new issues concerning interactions among group members.

Basic operational transform algorithms represent an established, proven approach to concurrent manipulation of data. These techniques are sufficiently general to support parallel production, real-time monitoring of dependent activities, and interdependent production tasks including annotation. While the fundamental characteristics of operational transform and related algorithms are data- and application-neutral, the details of their implementation are data-specific and user interface requirements are application-specific.

There are several remaining questions in terms of the applicability of operational transforms. An important concern is the amount of bandwidth that operational transform algorithms consume and how this bandwidth usage will change for yet to be developed applications. In particular, the algorithms have been shown to be effective for textual data and certain types of image data, but are refinements needed to support manipulation of more complex data? Another issue is the concept of undo, which is central to operational transforms. It is important to consider how well existing semantics for undo operations will scale or change to accommodate future data objects. In addition, new user interface techniques need to be developed to manage the distinction between undo of local and remote modifications. Finally, operational transform techniques inherently maintain information about which user has made each change, though this information is not generally preserved after conflict resolution. Can efficient techniques be developed to store attribution information without unnecessarily bloating the underlying data?

One of our goals is to make collaboration invisible, so that it is experienced as an integral part of production rather than a separate activity of its own. If collaboration is truly integrated into a group's activities, all work has the potential to be concurrent, thereby reducing the coordination overhead associated with collaborative production (Grudin 1994). Unfortunately achieving invisibility in collaboration raises yet another set of issues. One of these is data security and consistency. Invisibility means that group members have the ability to edit shared data. Thus an important issue is whether document management systems can be integrated in a way that provides recovery from mistakes and prevents potential collaborators from accidentally manipulating divergent copies of shared data. A second issue is whether different modes of undo support (e.g., at the level of individual objects, documents, projects) can be seamlessly integrated. It may be that supporting different ways to undo and redo behaviors will enable group members to interact more naturally, in both planned and *ad hoc* ways.

9.3 A Collaborative Production Scenario

In this section we present a scenario developed through interviews with NIMA employees; the scenario was developed to illustrate our vision for concurrent editing as a collaborative production task. It scenario shows how a distributed group of analysts might interact using technology to collaboratively produce a piece of imagery (a brief introduction to the scenario was used in Table 1). Interspersed between pieces of the scenario is commentary that describes important concepts.

> *The fall of a totalitarian regime eventually leads to a treaty establishing boundaries for three nations, largely along historical and ethnic boundaries. The team responsible for updating map data to include the new boundaries and other relevant features includes Anna and Bill, the analysts who provide an annotated translation of the treaty; Carla and Don, the analysts who will update the map data; and Ed, the manager ultimately responsible for the work.*

At the present, workflow in a project such as this would be fairly linear. Ed would likely monitor the progress of the translation task and tell Carla and Don to update the map data only after the translation is complete. Collaborative production, however, would allow group members to overlap effort making the workflow less linear and more efficient.

> *Anna has considerably more experience with this type of translation task and is largely serving as a mentor for Bill who will be performing the bulk of the work. They are located in the same building, but they have significantly different schedules and may or may not be working concurrently at any given time.*
>
> *Both translators access the source documents, the sections of the original treaty, and the translation from their desktops via a virtual shared disk managed by the document management system. The documents are opened in a standard COTS word processing package, and collaboration transparency services are automatically attached to support concurrent modifications within the same document. Awareness features integrated with the document management system and concurrent editing support maintain information about which users are currently accessing which sets of documents.*

As she does other work, Anna keeps an eye on Bill's entry in her user list and ensures that her on-line status indicators shows that she is available for quick questions should Bill run into any problems as he begins his translation work. Periodically she opens the document to check the details of his translation. Bill notices when she "arrives" and usually takes these opportunities to ask for help on problematic parts of the treaty.

In this case, the two translators are able to switch between simple division-of-labor and more complex opportunistic mentoring interactions. The awareness tools provide both explicit and implicit information about availability, while the transparent support for concurrent editing removes the overhead of coordinating document access or transfer. These interactions would be augmented by currently available communication channels including text chat, email, audio/visual conferencing, or telephone calls.

Ed, Carla, and Don have read- and annotate-only access to the "in progress" translation document as well as awareness information. They too look at the translated document as it is being constructed and notice that determining the locations of several of the border segments will require imagery that is more detailed than what they have on hand. Don points this out to Ed who begins searching for additional imagery sources.

Here, transparent concurrent editing allows elements of an otherwise linear process with dependencies to overlap, even though the users are not collaborating in the sense of mutually editing an artifact. Additional visibility across the workflow allows discovery of the requirement for new data much earlier in the process.

This kind of visibility could introduce numerous other subtle changes in the way that the team works. For example, the cartographers would be better able to plan because they would have the ability to make their own assessments of the progress of the translation and the magnitude of the changes that will be required. They would also have an opportunity to spot ambiguities earlier in the process before the translators have moved on to their next task.

Ed is able to locate more detailed imagery as well as some newly acquired commercial maps from a foreign company as the translation is finishing. He divides the region in half, assigning the halves to Carla and Don via email. Anna was copied on the email and adds annotations to the translated documents *indicating which parts are relevant to each cartographer and which describe border segments that span the two halves.*

This is a simple illustration of the ability to adapt intermediate deliverables as later workflow stages evolve. Anna knows that by adding some quick notes to the translation, she can save confusion later in the process. With transparent concurrent editing capabilities mediated through the underlying document management systems, she does not have to worry that Carla and Don might have already received divergent copies of the translation or even that they might be viewing the document as she is updating it.

Once the region assignments are made, the cartographers generally work concurrently. They open the map file from the document management system, and as with the word processing documents used by the translators, collaboration transparency services are automatically attached to standard COTS tools.

Don and Carla each occasionally scroll over to the other colleague's region, but they essentially work independently until they approach the boundary between their regions. They notice that one segment of the border that crosses the edge between the two sub-regions runs along a river, most of which is in Carla's region. Because she has already studied images of the river, Don suggests that she should cross the edge and complete the segment.

Even in situations with well-defined divisions of labor, increased awareness of other members' work allows potential conflicts to be negotiated as they emerge within the context of the work. Furthermore, transparent concurrent editing allows the division of work to be flexible, for example, by allowing one user to work in the space of data assigned to another user without the overhead of coordinating file access or merging.

As Carla is updating the border segment that spans the boundary between sub-regions, she inadvertently deletes a road segment that Don had just added. Don notices this, reviews the history of modifications made by both users, spots the feature deletion, and selects the "undo" operation. The road reappears, and subsequent modifications are not affected.

A primary risk introduced by concurrent editing capabilities is the possibility for destructive operations made either by accident or through misunderstandings. Addressing this problem requires development of usable solutions for both detection of problematic edits and recovery from those edits.

> *Bill notices that Carla is active and has the translated treaty document open. He opens the map that Don and Carla are updating and sees that border checkpoints have been added along the roads that connect the new countries. He notices, however, that a checkpoint is missing on a newly constructed road. Returning to the original treaty, he realizes that the omission was likely caused by an ambiguity in his translation, which had caused Anna to associate the relevant text with the wrong sub-region. He updates the translation, opens a chat session with Carla to make sure that she had not intentionally delayed adding the checkpoint for some other reason, and then sends a note to the group describing the change.*

Each collaborator is likely to have unique insights into the task, so allowing greater visibility of work in progress can allow missteps to be discovered and corrected earlier in the process. Integration with awareness and communication tools allows problems to be diagnosed and corrected in the context of the project's artifacts.

> *Once the map updates have been completed, all five team members meet online to review the results. Don spots an instance where a bridge appears to be on the wrong side of the border and opens and annotates an image to illustrate the problem. Anna adds a note to the translated document, and the group helps Ed compose an email with the relevant treaty segment and annotated image included to their State Department contact requesting clarification.*
>
> *Bill then points out a road that appears to be duplicated on the map. Ed scrolls through a history of the document and finds the version where the duplicate road was introduced. He notices that it appears to have been a simple copy/paste error, and he removes the duplicate.*

Online meetings are relatively well supported in current conferencing technologies. In this scenario, however, the meeting is simply a natural extension of existing work practices. The meeting only requires that they all access the relevant documents, ideally in the same manner that they would access the documents to do "normal" work together.

10 Concluding Remarks

Collaboration software continues to extend into new areas and create new challenges. For example, one area where collaboration is starting to become more important is visualization. Brewer, MacEachren, Abdo, Gundrum, and Otto (2000) and MacEachren (2000) evaluate the usability of tools and discuss current issues for same-time different place and same-place collaboration on geographic visualizations. As collaborative technologies continue to extend into new areas, the systems must be evaluated. At least once framework for evaluating systems in a low-cost manner already exists (Damianos *et al.* 1999). More evaluation and methods of evaluating, however, will be needed as we work toward the goal of invisible collaborative production.

Research is already moving in directions that should increase the invisibility of collaborative support. Agarwal, McParland, and Perry (2002) present a web-based tool that supports continuous and ad-hoc collaboration with a focus on day to day collaboration. Users are able to communicate through a variety of channels, share documents, track workflow, and activate videoconferences. The tool supports collaboration as an integral part of work and not as an activity that a user must consciously choose to initiate. Bernstein (2000) also describes related research, which discusses situated verses structured work and makes the argument for a hybrid approach that flexibly supports structured work. Our vision for collaborative editing is implicitly built upon this idea of flexibility. Along this line, the Command Post of the Future is a system based on the Multimodal Toolkit (MMI), which allows users to interact with an application through speech, handwriting, gestures, and gaze (Myers *et al.* 2002). This experimental system illustrates how increasing the variety of input types might help to make collaboration a more natural and inclusive activity.

References

Ackerman, M., D. Hindus, S. Mainwaring, and S. Starr, "Hanging on the wire: A field study of an audio-only media space," *ACM Transactions on Computer-Human Interaction*, 1997, 4:1, pp. 39-66.

Agarwal, D., C. McParland, and M. Perry, "Supporting Collaborative Computing and Interaction," *Proceedings of the Grace Hopper Celebration of Women in Computing 2002 Conference*, 2002, Vancouver, Canada.

Aoki, P.M., M. Romaine, M.H. Szymanski, J.D. Thornton, D. Wilson, *et al.*, "The Mad Hatter's Cocktail Party: A Social Mobile Audio Space Supporting Multiple Conversations," *Proc. ACM SIGCHI Conf. on Human Factors in Computing Systems (CHI '03)*, 2003, Fort Lauderdale, FL, USA.

Arrow, H., J.E. McGrath, and J.L. Berdahl, *Small Groups as Complex, Systems: Formation, Coordination, Development, and Adaptation*, Thousand Oaks, CA, USA: Sage, 2000.

Begole, J., M.B. Rosson, and C.A. Shaffer, "Flexible Collaboration Transparency: Supporting Worker Independence in Replicated Application-Sharing Systems," *ACM Transactions on Computer Human Interaction (TOCHI) archive*, 1999, 6:2.

Benford, S., C. Brown, G. Reynard, and C. Greenhalgh, "Shared spaces: transportation, artificiality, and spatiality," 1996, Boston, Massachusetts, USA.

Berket, K. and D. Agarwal, "Enabling Secure Ad-hoc Collaboration," *Proceedings of the Workshop on Advanced Collaborative Environments*, 2003, Seattle, WA, USA.

Bernstein, A., "How Can Cooperative Work Tools Support Dynamic Group Processes? Bridging the Specificity Frontier," *Proceedings of Computer Supported Cooperative Work*, 2000, Philadelphia, PA, USA.

Billinghurst, M. and H. Kato, "Collaborative Mixed Reality," *Proceedings of International Symposium on Mixed Reality*, 1999, Yokohama, Japan.

Blum, R. and C. Gutwin, "GroupPlan (in development)," Available at: http://hci.usask.ca/projects/groupplan.xml.

Boyle, M., C. Edwards, and S. Greenberg, "The Effects of Filtered Video on Awareness and Privacy," *Proceedings of the CSCW'00 Conference on Computer Supported Cooperative Work*, 2000, Philadelphia, Pennsylvania, USA.

Bradner, E., W.A. Kellogg, and T. Erickson, "The Adoption and Use of BABBLE: A Field Study of Chat in the Workplace," *Proc. European Conf. Computer-Supported Cooperative Work ECSCW'99*, 1999, Copenhagen, Denmark.

Breese, J., D. Heckerman, and C. Kadie, "Empirical Analysis of Predictive Algorithms for Collaborative Filtering," *Proceedings of the 14th Conference on Uncertainty in Artifical Intelligence*, 1998, Madison, WI, USA.

Brewer, I., A.M. MacEachren, H. Abdo, J. Gundrum, and G. Otto, "Collorative Geographic Visualization: Enabling Shared Understanding of Environmental Processes," *IEEE Information Visualization Symposium*, 2000, Salt Lake City, Utah, USA.

Carroll, J.M., D.C. Neale, P.L. Isenhour, M.B. Rosson, and D.S. McCrickard, "Notification and awareness: Synchronizing task-oriented collaborative activity," *International Journal of Human Computer Studies*, 2003, 58:5, pp. 605-632.

Chang, B.-W., "In-place editing of web pages: Sparrow community-shared documents," *Seventh World Wide Web Conference*, 1998.

Choudhary, R. and P. Dewan, "A General Multi-User Undo/Redo Model," *Proceedings of European Conference on Computer Supported Work*, 1995.

Chung, G. and P. Dewan, "Towards dynamic collaboration architectures," *Computer Supported Cooperative Work*, 2004, Chicago, Illinois, USA.

Daft, R.L. and R.H. Lengel, "Organizational Information Requirements, Media Richness and Structural Design," *Management Science*, 1986, 32:5, pp. 554-571.

Damianos, L., L. Hirschman, R. Kozierok, J. Kurtz, A. Greenberg, *et al.*, "Evaluation for Collaborative Systems," *ACM Computing Surveys (CSUR)*, 1999, 31:2, pp. 17 (article 15).

Dennis, A.R. and J.S. Valacich, "Rethinking Media Richness: Towards a Theory of Media Synchronicity," *Proceedings of Hawaii International Conference on System Sciences*, 1999, Maui, HI, USA.

Dewan, P., "Architectures for Collaborative Applications," in *Computer Supported Cooperative Work*, Beaudouin-Lafon (ed.), John Wiley and Sons, 1999.

Dewan, P. and A. Sharma, "An Experiment in Interoperating Heterogeneous Collaborative Systems," *European Conference on Computer Supported Cooperative Work*, 1999.

Dourish, P. and V. Bellotti, "Awareness and Coordination in Shared Workspaces," *Proceedings of Computer Supported Cooperative Work*, 1992, Toronto, Canada.

Dyck, J. and C. Gutwin, "Groupspace: A 3D Workspace Supporting User Awareness," *Companion Proceedings of the ACM Conference on Human Factors in Computing Systems*, 2002, Minneapolis, MN, USA.

Ellis, S., J. Gibbs, and G.L. Rein, "GroupWare: Some Issues and Experiences," *Communications of the ACM*, 1991, 34:1, pp. 38-58.

eRoom, Available at: http://www.documentum.com/products/glossary/eroom.htm.

Ferscha, A., C. Holzmann, and S. Oppl, "Context awareness for group interaction support," *Conference on Mobile Computing and Networking*, 2004, Philadelphia, PA, USA.

Greenberg, S., "Collaborative Interfaces for the Web," in *Human Factors and Web Development*, C. Forsythe, E. Grose and J. Ratner eds.), LEA Press, 1997.

Greenberg, S. and H. Kuzuoka, "Using Digital but Physical Surrogates to Mediate Awareness, Communication and Privacy in Media Spaces," *Personal Technologies*, 2000, 4:1.

Greenberg, S. and M. Rounding, "The Notification Collage: Posting Information to Public and Personal Displays," *Proceedings of the ACM Conference on Human Factors in Computing Systems [CHI Letters 3(1)]*, 2000.

Groove, Available at: http://www.groove.net/.

Gross, T., "CYCLADES: A Distributed System for Virtual Community Support Based on Open Archives," *Proceedings of the Eleventh Euromicro Conference on Parallel, Distributed, and Network-Based Processing*, 2003, Genova, Italy.

Grudin, J., "Groupware and Social Dynamics: Eight Challenges for Developers," *Communications of the ACM*, 1994, 37:1, pp. 92-105.

Gutwin, C. and S. Greenberg, "The Effects of Workspace Awareness Support on the Usability of Real-Time Distributed Groupware," *ACM Transactions on Computer-Human Interaction (TOCHI)*, 1999, 6:3, pp. 243-281.

Hall, R.W., A. Mathur, F. Jahanian, A. Prakash, and C. Rassmussen, "Corona: A Communication Service for Scalable, Reliable Group Collaboration Systems," *Proceedings of the 1996 ACM conference on Computer Supported Cooperative Work*, 1996, Boston, Massachusetts, USA.

Horvitz, E., A. Jacobs, and D. Hovel, "Attention-Sensing Alerting," *Proceedings of UAI'99, Conference on Uncertainty and Artificial Intelligence*, 1999, Stockholm, Sweden.

InfoWorkSpace, Available at: http://www.InfoWorkSpace.com.

Ionescu, M., A.M. Krebs, and I. Marsic, "Dynamic Content and Offline Collaboration in Synchronous Groupware," *Proceedings of the Collaborative Technologies Symposium*, 2002, San Antonio, TX, USA.

Jackson, L.S. and E. Grossman, "Integration of Synchronous and Asynchronous Collaboration Activities," *ACM Computing Surveys (CSUR)*, 1999, 31:2, pp. 5 (article 12).

Kermarrec, A.M., A. Rowstron, M. Shapiro, and P. Druschel, "The IceCube Approach to the Reconciliation of Diverging Replicas," *Proceedings of the 20th annual ACM SIGACT-SIGOPS Symposium on Priciples of Distributed Computing*, 2001, Newport, R.I., USA.

Krebs, A.M., M. Ionescu, B. Dorohonceanu, and I. Marsic, "The DISCIPLE System for Collaboration Over the Heterogeneous Web," *Proceedings of the 36th Hawaiian International Conference on System Sciences*, 2003, Waikoloa, Big Island, Hawaii, USA.

Laurillau, Y. and L. Nigay, "Clover Architecture for Groupware," *Computer Supported Cooperative Work (CSCW)*, 2002, New Orleans, Louisiana, USA.

Li, D., "Sharing Single-User Editors by Intelligent Collaboration transparency," *The third annual international workshop on collaborative editing systems (IWCES-3), jointly with ACM SIGGROUP'01 Conference*, 2001, Boulder, Colorado, USA.

Li, D. and R. Li, "Transparent Sharing and Interoperation of Heterogeneous Single-User Applications," *Computer Supported Cooperative Work (CSCW)*, 2002, New Orleans, Louisiana, USA.

Li, D. and J. Patrao, "Demonstrational Customization of a Shared Whiteboard to Support User-Defined Semantic Relationships among Objects," *ACM SIGGROUP'01 Conference on Supporting Group Work*, 2001, Boulder, Colorado, USA.

Li, R. and D. Li, "A Regional Undo Mechanism for Text Editing," *The 5th International Workshop on Collaborative Editing Systems (IWCES-5)*, 2003.

Linebarger, J.M. and G.D. Kessler, "GroupMorph: Supporting Group Collaboration Models for Hierarchically Decomposable Tasks in 3D Environments," *ACM Conference on Computer Supported Collaborative Work, Poster Session*, 2002, New Orleans, LA, USA.

MacEachren, A.M., "Cartography and GIS: Facilitating Collaboration," *Progress in Human Geograhpy*, 2000, 24:3, pp. 445-456.

Malone, T.W. and K. Crowston, "The Interdisciplinary Study of Coordination," *ACM Computing Surveys (CSUR)*, 1994, 26:1, pp. 87-119.

Mark, G., L. Fuchs, and M. Sohlenkamp, "Supporting groupware conventions through contextual awareness," *Proceedings of ECSCW'97*, 1997, Lancaster, England.

McCrickard, S., "Notification User Interfaces," *International Journal of Human Computer Studies*, 2003, 58:5.

McDonald, D.W., C. Weng, and J.H. Gennari, "The multiple views of inter-organizational authoring," *Computer Supported Cooperative Work*, 2004, Chicago, Illinois, USA.

McGrath, J.E., "Time, Interaction, and Performance (TIP): A Theory of Groups," *Small Group Res.*, 1991, 22, pp. 147-174.

Mills, K.L., "Introduction to the Electronic Symposium on Computer-Supported Cooperative Work," *ACM Computing Surveys (CSUR)*, 1999, 31:2, pp. 105-115.

Myers, B., R. Malkin, M. Bett, A. Waibel, B. Bostwick, *et al.*, "Flexi-Modal and Multi-Machine User Interfaces," *Proceedings of the Fourth IEEE International Conference on Multimodal Interfaces*, 2002, Pittsburgh, PA, USA.

NetMeeting, Available at: http://www.microsoft.com/windows/netmeeting/.

Neustaedter, C., "A 3D Instant Messenger Visualization Using a Space Metaphor," Alberta, Canada, Department of Computer Science, University of Calgary, 2001.

O'Mahony, M., N. Hurley, N. Kushmerik, and G. Silvestre, "Collaborative recommendation: A robustness analysis," *ACM Transactions on Internet Technology*, 2004, 4:4, pp. 344-377.

Olson, G.M. and J.S. Olson, "Distance Matters," *Human Computer Interaction*, 2000, 15:2&3, pp. 139-179.

Orlikowski, W.J., "Learning from Notes: Organizational Issues in Groupware Implementation," *Proceedings of the Conference on Computer Supported Cooperative Work*, 1992, Toronto, Canada.

Parvin, B., J. Taylor, G. Cong, M. Okeefe, and M. Barcellos-Hoff, "DeepView: A Channel for Distributed Microscopy and Informatics," *IEEE Conference on HIgh Perofmrance Computing and Networking*, 1999.

Payton, D., M. Daily, and K. Martin, "Dynamic Collaborator Discovery in Information Intensive Environments," *ACM Computing Surveys (CSUR)*, 1999, 31:2, pp. 8 (article 8).

Penner, R. and C. Gutwin, "Collide "COLLaborative Integrated Software Development Environment"," Available at: http://hci.usask.ca/projects/collide.xml.

Perry, M. and D. Agarwal, "Remote Control for Videoconferencing," *Proceedings of the 11th International Conference of the Information Resources Management Association*, 2000, Anchorage, Alaska, USA.

Perry, M., D. Agarwal, and C. McParland, "Use Scenarios for Collaborative Editing in Scientific Collaborations," *4th International Workshop on Collaborative Editing, CSCW 2002 Conference*, 2002, New Orleans, Louisiana, USA.

Prakash, A. and H.S. Shim, "DistView: Support for Building Efficient Collaborative Applications Using Replicated Objects," *Proceedings of the 1994 ACM Conference on Computer Supported Cooperative WOrk*, 1994, Chapel Hill, North Carolina, USA.

Presence-AR, Available at: http://www.advancedreality.com.

Prinz, W., W. Grather, T. Gross, K.H. Klein, S. Kolvenbach, *et al.*, "Prasentation von Aktivitaten bei verteilter Zusammenarbeit (in German)," *Mensch & Computer, Fachubergreifende Konferenz, Vom interaktiven Werkzeug zu kooperativen Arbeits und Lernwelten*, 2002, Hamburg, Germany.

Rees, J., S. Ferguson, and S. Virdhagriswaran, "Consistency Management for Distributed Collaboration," *ACM Computing Surveys (CSUR)*, 1999, 31:2, pp. 3 (article 13).

Roberts-Witt, S.L., *A "Eureka!" Moment at Xerox*.

Roseman, M. and S. Greenberg, "Building Real Time Groupware with GroupKit, A Groupware Toolkit," *ACM Transactions on Computer Human Interaction*, 1996, 3:1, pp. 66-106.

Roseman, M. and S. Greenberg, *A Tour of TeamRooms*.

Schmidt, K. and L. Bannon, "Taking CSCW seriously: Supporting articulation work," *Computer-Supported Cooperative Work*, 1992, 1:1, pp. 7-40.

Schmidt, K. and C. Simone, "Coordination mechanisms: Toward a conceptual foundation of CSCW systems design," *Computer-Supported Cooperative Work*, 1996, 5:2, pp. 155-200.

Shen, H. and C. Sun, "Flexible Merging for Asynchronous Collaborative Systems," *Proceedings of the Tenth International Conference on Cooperative Information Systems*, 2002, University of California, Irvine, USA.

Smith, R.B., M. Wolczko, and D. Ungar, "From Kansas to Oz: Collaborative Debugging When a Shared World Breaks," *Communications of the ACM archive*, 1997, 40:4, pp. 72-78.

Sproul, L. and S. Kiesler, *Connections: New Ways of Working in the Networked Organization*, Cambridge, MA: MIT Press, 1991.

Suchman, L. and R. Trigg, "SESSION: Session VI - Collaboration Research Table of Contents," *Proceedings of the 1986 ACM Conference on Computer-Supported Cooperative Work*, 1986, Austin, Texas, USA.

Sun, C., "Undo as Concurrent Inverse in Group Editors," *ACM Transactions on Computer Human Interaction (TOCHI)*, 2002, 9:4, pp. 309-361.

Sun, C. and D. Chen, "Consistency Maintenance in Real-Time Collaborative Graphics Editing Systems," *ACM Transactions on Computer Human Interaction (TOCHI)*, 2002, 9:1, pp. 1-41.

Sun, D., S. Xia, C. Sun, and D. Chen, "Operational transformation for collaborative word processing," *Computer Supported Cooperative Work*, 2004, Chicago, Illinois, USA.

Tam, J., L. McCaffrey, F. Maurer, and S. Greenberg, "Change Awareness in Software Engineering Using Two Dimensional Graphical Design and Development Tools," Alberta, Canada, Department of Computer Science, University of Calgary, 2000.

Tang, J., N. Yankelovich, J. Begole, M. Van Kleek, F. Li, *et al.*, "ConNexus to Awarenex: Extending Awareness to Mobile Users," *CHI 2001*, 2001, Seattle, Washington, USA.

Velez, M., M.M. Tremaine, A. Sarcevic, B. Dorohonceanu, A. Krebs, *et al.*, ""Who's in charge here?" communicating across unequal computer platforms," *ACM Trans. Comput.-Hum. Interact.*, 2004, 11:4, pp. 407-444.

Wang, X., J. Bu, and C. Chen, "Achieving Undo in Bitmap-Based Collaborative Graphics Editing Systems," *Computer Supported Cooperative Work (CSCW)*, 2002, New Orleans, Louisiana, USA.

Wolverton, M., "Task-based Information Management," *ACM Computing Surveys (CSUR)*, 1999, 31:2, pp. 8 (article 10).

Xia, S., D. Sun, C. Sun, D. Chen, and H. Shen, *Leveraging single-user applications for multi-user collaboration: the Coword approach*, Chicago, Illinois, USA: ACM Press, 2004.

Yang, Y., P. Inbarajan, and D. Li, "Evolving Framework, Evolving Groupware: A Component-Based Approach to Designing for Moving Targets," Texas A&M, 2003.