Preface

The focus on innovation in today's business world has increased awareness of the crucial role collaboration plays in the workplace. At the same time, business globalization often leads to the geographical separation of teams within an organization, which makes collaboration more difficult. Although collaboration software can bridge time and distance, enabling people to work together in these new situations, for many work processes it brings a lack of flexibility which stifles creativity and innovation. The need to accommodate both formal and informal collaboration has been at the core of the interdisciplinary field of computer-supported cooperative work since it emerged in the mid-80s as a specialty, in part as a reaction to trends in office automation toward formal workflow. The collaboration researchers at IBM Research who contributed to this issue are all part of this tradition, grounding their work in user studies, doing iterative design based on user reactions, and perhaps most important, striving to find ways to support both the formal aspects of work and the informal, inspirational parts.

The papers in this issue are organized in a sequence that starts with a broad overview of business activities, followed by a number of loosely related papers on other aspects of collaboration: an activity-based model of collaboration, two implementation papers on collaboration software, two papers on user studies, and two additional papers that deal with analytics and visualization in support of collaboration. Overall, these represent a sampling of collaboration-related research at IBM, with clear emphasis on business activities. An additional, non-topical paper deals with the use of ontology in the development of software architecture.

In their paper "Beyond predictable workflows: Enhancing productivity in artful business processes," Hill et al. use the term "artful processes" to describe a kind of usable workflow that really could succeed in enterprises. Their concept was developed based on user studies, test deployments, and close interactions with project teams working on new forms of e-mail, calendars, and activity management tools. It is a product team perspective and represents the key principles on which the team is basing its introduction of activity-centered collaboration to customers. The authors cite the researchers that influenced the product, referring to other papers in this issue and reflecting the close connections to specific project teams.

The paper by Moody et al., "Business activity patterns: A new model for collaborative business applications," outlines the principal goals of the Unified Activity Management project in the IBM Research Division, which range from enabling the user to organize all work items related to a given activity by directly linking all the relevant tools, to identifying work practice patterns in ad hoc activities. When ad hoc activities can be recorded and analyzed and even refined by those who apply a systematic approach to learning from experience, the resulting "activity pattern" can be used as a template for initiating instances of that type of activity. Activity patterns can be used to represent best practices in an organization—in this form, they provide guidance to those starting out on the activity, in a form in which any ad hoc communication used to make things work can be added and preserved in context. These activity patterns, which can be continually edited and improved,

0018-8670/06/$5.00

represent a facility for organizational learning and expertise sharing.

The paper by Cozzi et al., "Activity management as a Web service," a companion to the paper by Moody et al., describes one of the first attempts to make activity support an open service, decoupled from Lotus* Workplace* products but still able to interact with Lotus documents, other content repositories, Web pages, instant messages, and e-mail. Its implementation is based on the underlying Resource Description Framework representation of activities, which the team expects can eventually represent both formal and informal workflows. The team was able to develop a number of different user interfaces to the same data, which speaks well for the potential applicability and extensibility of activity support as an open service. Probably even more important though is the use of REST/XML APIs, which facilitates Web-2.0-style "mash-ups." This has led, within a limited internal deployment, to a number of interesting new capabilities in which activities have been integrated with other applications, such as e-mail and dashboards for process monitoring.

Activity Explorer (AE) is the first activity-centered collaboration feature set to emerge in a Lotus product, having shipped in IBM Workplace Managed Client* V2.6. In the paper "Activity Explorer: Activity-centric collaboration from research to product," Geyer et al. recount how ideas developed from analysis of e-mail initially, then through iterative design with user teams, and on to implementation by the product team. Having a base set of features in the product has facilitated experimentation as well, as there have been extensions in support of patterns and additional user testing that has shaped the ongoing research program. AE includes notions of document awareness as well as people awareness. To avoid an anticipated, overly intrusive set of interruptions, the AE team has expanded its focus to include attention management and notification about new activities. Some of the user interface ideas, including activity threads and openness concepts discussed in the preceding papers, continue to influence design of follow-on products, the core service being known as Open Activities, now adopted by the Lotus product team.

Although SCOUT, the system described by Sow et al. in their paper "Uncovering the to-dos hidden in your in-box," has not been formally part of the activity-centered collaboration research program, it provides some important complementary technology for managing workflow. The to-dos in the in-box consist of machine-generated e-mail notifications that employees regularly receive from enterprise workflow systems. Most are easily recognized by using text analytics, making it possible to get system support for managing them. It is easy to imagine taking these parsed e-mails into an activity framework, making it possible for an activity support system to interact with pre-existing workflow systems that generate these emails. SCOUT's ability to recognize a set of machine-generated e-mail messages, to extract from them task information, and to display this information in a context-based graphical interface has been demonstrated in a preliminary evaluation study.

The work described in the paper "Ethnographic study of collaborative knowledge work" by Kogan and Muller, which was part of the Unified Activity Management project, attempts to identify the type of support that knowledge workers could benefit from in the performance of their tasks in a collaborative environment. The authors present the results of a field study that illustrates the complex, dynamic work environments in which knowledge workers are compelled to develop individualized, ad hoc work processes in order to get their work done.

Geographical separation measured in time zones is becoming a serious impediment to the performance of collaborating teams. In the paper "Following the sun: Case studies in global software development," Treinen and Miller-Frost present two case studies in which they examine whether it is possible to create a development environment in which tasks are handed off from teams ending their work day to those beginning their shifts elsewhere in the world. The authors identify a number of conditions essential to making round-the-clock development possible. Many of these are cultural and have to do with clarity in goals and communication at hand-off points. Although the work was not done with activities in mind, it is interesting to speculate on how an activities framework could address some of these issues.

Information originating in electronic forums, chats, and blogs are increasingly of interest to businesses because they provide indications of trends and patterns in communication involving employees and

customers. Existing data-mining techniques are not quite suitable for these unstructured data. In their paper "Machines in the conversation: Detecting themes and trends in informal communication streams," Spangler, Kreulen, and Newswanger describe techniques developed for precisely these types of communication streams. They applied these techniques to so-called "Jam" events, corporate-wide sessions during which tens of thousands of IBM employees brainstormed on a topic of interest.

Whereas in the past, work on visualization focused on the individual user, the Technical Forum article "Communication-minded visualization: A call to action" by Viégas and Wattenberg proposes a new research direction that is intended to expand this outlook to include collaborative environments. The authors, who refer to this direction as communication-minded visualization, offer convincing evidence of the importance of this heretofore neglected area of research.

The area addressed by the last paper lies outside the business collaboration theme of this issue. In their paper "Using ontology to support development of software architectures," Akerman and Tyree advocate an approach to software development that is based on architectural decisions and in which the architecture is captured by an instance of an ontology. The authors, practitioners themselves, demonstrate the applicability of this approach by describing the design of a real-time credit approval system. The implementation of this application is supported by the use of Protégé, an open-source ontology development tool.

The next issue of the *Journal* is devoted to information-based medicine.

Irene Greif, IBM Fellow and Department Group Manager, Collaborative User Experience
IBM Research Division

Alex Birman, Associate Editor

John J. Ritsko, Editor-in-Chief