Supporting Collaborative Computing and Interaction*

Deborah Agarwal, Charles McParland, and Marcia Perry
Distributed Systems Department, Lawrence Berkeley National Laboratory
DAAgarwal@lbl.gov, CPMcparland@lbl.gov, Mperry@lbl.gov

Abstract

To enable collaboration on the daily tasks involved in scientific research, collaborative frameworks should provide lightweight and ubiquitous components that support a wide variety of interaction modes. We envision a collaborative environment as one that provides a persistent space within which participants can locate each other, exchange synchronous and asynchronous messages, share documents and applications, share workflow, and hold videoconferences. We are developing the Pervasive Collaborative Computing Environment (PCCE) as such an environment. The PCCE will provide integrated tools to support shared computing and task control and monitoring. This paper describes the PCCE and the rationale for its design.

1 Introduction

Scientific collaborations are organized to accomplish a wide variety of tasks within a wide range of organizational structure and formality. Within this domain, we find highly structured environments (e.g., the production analysis phase of high-energy physics experiments) as well as informal, spontaneous collaborations (e.g., co-authoring, cooperative software development, and instrument "debugging" efforts). In our experience, the present lack of an integrated set of software tools to facilitate the broad range of collaborative interactions discourages the formation of effective collaborative teams.

Many of today's scientific collaboratory tools such as videoconferencing tools are highly interactive and only support formal meetings well. Although videoconferencing is an important part of a collaborative environment, much of the work of scientific collaboration requires more informal and asynchronous mechanisms. Collaborators also need light-weight and flexible ways to stay in touch. When collaborations are world-wide and often extend beyond normal working hours or have limited satellite operations, asynchronous communication becomes critical. Collaborators currently revert to email and telephone because present software lacks asynchronous, ad hoc communication mechanisms. However, email messages are often delivered out of sequence and responses from collaborators can become disorganized. Also, the typical scientist now receives so much email during the course of a day that it is generally ignored during periods of concentrated work and this is often when it is most critical for collaborators to communicate.

Engaging in informal interactions and sharing documents and data have been shown to be an important part of an effective collaboration[16]. There is a need for collaborative tools that support connecting people so that they feel like they are working together on a daily, hourly, or even continuous basis [2][4]. Messaging systems and presence awareness mechanisms can help to make collaborations successful because they provide basic connectivity in a shared context that allows collaborators to hold spontaneous meetings[13].

Computing activities, including shared program development and debugging and 24x7 data analysis operations, have become a central feature of modern scientific and engineering research efforts. These efforts are increasingly collaborative, involving individuals and institutions at locations distributed all over the globe. Grids[8] provide consistent, secure access to distributed computing resources and have gained widespread acceptance as an important future architecture for distributed computing. Given the central role of computing in almost every facet of daily scientific research life, grids are becoming the primary environment within which distributed users share computing tasks, analyses, and visualization results. While existing collaboration tools can help mediate some of the communications needs of this environment, the grid brings additional security requirements and opportunities for the development of tools that interact with computations.

* This work was supported by the Director, Office of Science, Office of Advanced Computing Research, of the U.S. Department of Energy under Contract No. DE-AC03-76SF00098. See disclaimer at http://www-library.lbl.gov/disclaimer. LBNL report number LBNL-50418.

2 Background

Many tools have been developed for online collaboration and in this section we review some of the tools relevant to scientific collaboration. These include tools and systems for messaging, on-line instrument access, and videoconferencing. These tools tend to target specific interaction modes rather than a continuum of interaction paradigms.

Among the text-based messaging systems that have been developed over the past several years are the Xerox PARC MOO[6] and the Internet Relay Chat (IRC)[19]. The MOO provides persistent virtual rooms that users can create and move between for exchanging messages with other occupants. The IRC metaphor for rooms is channels; anyone can create a channel and it only exists while someone is in the channel. Recent systems such as the America On-Line Instant Messaging program (AIM)[20] and the ICQ messaging system[21] are primarily intended for one-to-one conversation. Later versions of AIM include group chat, file transmit, and voice capabilities. Businesses have begun to use messaging tools for informal communication in the workplace. Recent research efforts[13] have found that the use of these tools contributes to a sense of co-worker community often lacking in a large distributed workplace. The tools are also seen as alleviating some of the isolation and fragmentation that accompanies telecommuting or constant travel. Lotus and Microsoft have developed their own versions of the instant message service and the Lotus system supports secure private messaging between authenticated users.

Several systems such as the Virtual Room Videoconferencing System (VRVS)[22] and WebEx[23] offer videoconferencing capabilities. These tools provide conventional two-way videoconferencing and broadcast of one primary video and audio feed to an audience. The MBone videoconferencing tools[24] (vic, vat, rat, wb, and sdr)[14] provide multicast-based, multi-way videoconferencing that allows all users to be seen and heard as equal participants in the videoconference. The Access Grid[25] has expanded on this idea by providing relatively natural group-to-group interaction capabilities and launching mechanisms that allow many groups to participate simultaneously. Recently the VRVS system has been enhanced to include interaction with MBone and Access Grid videoconferencing sessions.

One of the first collaboration frameworks developed was Habanero[10]. upon which the CORE2000 remote scientific instrument access system[26] is based. The Collaborative Virtual Workspace (CVW) is a prototype collaborative interaction environment that supports the ongoing, daily, or ad hoc collaborative needs of geographically and temporally dispersed work groups[27]. CVW provides users with a persistent virtual space in which they can share documents, data, and applications, exchange synchronous and asynchronous messages, and use the MBone videoconferencing tools. The Basic Support for Collaborative Work (BSCW)[28] system provides web-based spaces for file-sharing and discussion.

Groove[29] facilitates group work by supporting instant messaging, file-sharing, web-browsing, and forum discussions. As a composable framework, it offers a set of customizable tools. Based on a peer-to-peer paradigm, there is no need for a centralized server. Groove also supports offline use, message encryption, and persistence across computers.

The US Department of Energy has sponsored several collaboratory tool research and development projects including development of an electronic notebook[30], a collaboratory interoperability framework[1], the CORE2000 system, the Akenti authorization server[15], and Access Grid components. Concurrent with development of these underlying technologies, DOE has also sponsored the development of several pilot collaboratories in specific scientific disciplines including a virtual NMR facility[11], the Materials Microcharacterization Collaboratory[17], the Diesel Combustion Collaboratory[31], and the SpectroMicroscopy Collaboratory[4][32]. These early projects provided important insights into the scientific collaboration environment.

The various collaboratory components and tools available today provide a stepping-stone to the capabilities we need for successful collaboratories. In the next section, we introduce our view of how the existing technologies should be employed to create a collaborative computing capability.

3 Characteristics of a Collaboration Framework

Existing research and our experience building collaboratories have shown that an effective collaboration framework should provide a consistent environment that offers inviting and tangible benefits to all its users. The most fundamental characteristic of a collaborative environment is ubiquity. Collaborators should be able to enter and work within the environment from their desktop machine, their laptop, another user's computer, or any other digital device. They should also be able to participate from any location (e.g. airport, home, etc.).

Collaboration tools need to be activated in the context of the day-to-day activities of the participants. When collaboration tools are a natural part of working on the computer, a critical mass of collaborators is created in the environment. This makes it possible to find other collaborators easily. If only a subset of the participants login regularly, or if the tools are launched in an out-of-context separate step, collaborators will use them only when they want to contact another user and will often not find them in the environment. The collaborative tools need to provide a real benefit to all the users; a one-sided benefit model is counterproductive in a collaboration environment[9].

While the exact set of collaboratory tools available to the user will depend on the equipment and computing resources available at the user's immediate location, all users should have access to a common core set of collaboratory tools. This core set should provide both the sense and the benefits of connection to the collaboration space regardless of the user's physical location. The minimum set of capabilities should include a directory service, a contact mechanism, and a text-based messaging capability. The minimum user interface should be a web browser.

Furthermore, collaboration members should have reasonable assurance of the identity of participants who are present within the collaboration space. During face-to-face interactions, free and open communication is possible because participants use visual and aural "clues" to recognize eachother. Although these "clues" are lacking from many types of electronic communication (e.g. email and instant messaging), knowing the participants' identities remains an important factor in comfort level during interactions. Collaboration tools will only succeed if they can assure participants of the identities of those with whom they are interacting and provide reasonable expectations of privacy during these interactions. Thus, security is another important characteristic of a collaborative environment.

A successful collaborative framework should be adaptable, allowing participants to configure the mode, length and scope of their interactions to meet their own needs. The environment should provide a tangible sense of persistence, which promotes the feeling that a user has returned to a familiar place. Shared virtual meeting places with their documents and other contents should be as they were left, with document and directory displays in the same format. In addition, each collaborator should have a personal space where they can store things and carry them throughout the environment. The collaborative tools themselves should be invoked with the same configuration as when they were last used. Users should feel that they are returning to the same place in the virtual environment. In addition, the collaboration environment should work equally well for a small group of collaborators as for a large group. Generally, a large collaboration is built from a small group who has convinced others to use the tools. If the tools require a great deal of infrastructure to support them, it is unlikely that the small group would ever have managed to use the tools in the first place.

In normal group interactions, some issues are settled during informal or impromptu meetings, while other problems are only resolved at formal, scheduled meetings. This variability in human interactions cannot be designed away; it is integral to the way people interact. Therefore, a collaborative framework must support and facilitate the escalation and differentiation of communication behaviors. A collaborative framework must support and facilitate a continuum of communication services through an integrated interface so users can easily progress through the continuum and participate in conversations in whatever mode is appropriate at the time. This might mean messaging asynchronously, using an instant messaging system, having an audio conversation, holding a videoconference, sharing an application, or sharing data.

4 The Pervasive Collaborative Computing Environment (PCCE)

As can be seen from the preceding discussion, building a collaboration framework is a non-trivial project. One particularly difficult task is integrating the collaboration tools with the day-to-day scientific activities of the participants. This goal is difficult to achieve when building tools for use across a broad spectrum of sciences. In the PCCE we are focusing on building collaboration capabilities for activities that are heavily computation dependent. This allows us to concentrate on building a relatively small set of tools that can be immediately useful to scientists. The tools that we have identified as important in a computational environment include directory services, file sharing capabilities, text-based messaging, workflow management tools, and job submission and tracking mechanisms.

The underlying core component of the PCCE system is the PCCE server. The PCCE server is the initial login point for all users entering the collaboration. This server keeps track of the set of users authorized to join the collaboration and the set of users currently in the collaboration. It provides the directory services and the base connectivity to allow collaborators to see who is in the environment and contact each other. The PCCE server keeps track of the tools available in the environment and the correct information for contacting these tools. This server is enhanced with Java servlets to facilitate its use in a web-based environment. For example, the lightweight servlet processes coordinate login, authentication, and tool-launch information for an HTTP session so that the login process for browser-based clients can be completed in the background without additional user input. In addition, since software state can be maintained on the central server, users can be provided with persistence from session to session or from site to site. This server also facilitates the sense that the collaborative environment is the same on the laptop at a conference as it is on the desktop at the research site.

Another core component of the PCCE is text-based messaging provided by an IRC server and client. Normally the IRC protocol is implemented using unsecured connections and simple clear text password authentication of participants. We have written our own client and enhanced the IRC to provide SSL connections that allow for PKI authentication and message encryption. We have also added the ability to have persistent channels to provide rendezvous locations for particular interest groups. A nice feature of the IRC is the ability to be in multiple channels at once. This allows a participant to maintain a presence in the channels of interest and carry on private conversations with other participants simultaneously. We are also augmenting the IRC with asynchronous messaging capabilities to allow users to carry on conversations across logins and to allow users to leave messages for participants who are not currently logged in.

Within collaborations that support large computing efforts, there is a clear need for collaborative computational workflow management tools. While scientific research efforts will ultimately benefit from the development of extensive, fully-integrated problem solving environments, there is an immediate need for tool sets that facilitate the basic shared monitoring, control and scheduling operations found in most collaborations' computing activities. Presently the operational activities associated with computing remain outside of the collaborative mainstream and are thus difficult or impossible to share in any effective way. We feel that web-based tools that allow shared control of grid-based computing activities will increase a collaboration's productivity and enhance the sense of task sharing among its members.

File sharing is important for allowing participants to share information with each other. We are currently evaluating possible file-sharing solutions and are experimenting with incorporating the BSCW file sharing system in the PCCE to provide a file-sharing capability. This is an essential asynchronous collaboration capability.

We are integrating the Grid job submission and tracking mechanisms into the PCCE and augmenting them with emerging standards like WebDAV and web services to create a relatively lightweight, compute-centered workflow tracking system. The job submission and tracking mechanisms will be provided by Condor[33], which allows a set of compute jobs to be represented as a directed acyclic graph (DAG). The edges in the DAG represent the dependencies between the jobs and the DAG itself can be used to submit and track the jobs. The interface to this workflow system will include WebDAV tools and web services.

To minimize deployment and access issues, we are providing the PCCE through a web-based interface comprised of browser plug-ins, applets, and Java beans. This user interface allows access to the collaborative environment by entering a URL and provides a consistent look and feel across different platforms. The web-based model of deployment allows the software programming effort to focus on development instead of distribution. Versions of the tools that run outside the browser will also be available as stand-alone applications so they can be integrated with other applications.

Since the natural development of a collaboration is usually incremental in nature and builds slowly over time, we see a need for increased peer-to-peer architectures in collaborative frameworks to support scaling to both small and large collaborations. The genesis of a collaboration is often just two to three people working together on a specific short-term problem. The small group often slowly draws other people in because they perceive a benefit to joining the collaboration. A server-based architecture makes it difficult to support incremental building of the collaboration since installation, maintenance and support of servers requires significant effort and is beyond the scope of a small, ad hoc collaboration. A peer-to-peer architecture and framework is a relatively ideal mechanism to support incremental building of a collaboration group. We feel that a reliable and secure group communication system is the best mechanism for achieving scalability, peer-to-peer capabilities, and security, without requiring centralized servers. Instead, the servers provide added value capabilities such as archiving and persistence information.

We plan to integrate the InterGroup reliable multicast protocol[5] to provide efficient group communication capabilities. InterGroup provides a variety of group oriented message delivery services and it is designed to be efficient and scalable to large groups of collaborators. The Secure Group Layer (SGL) is designed to operate over InterGroup and provide security services to the group. These security services are similar to those provided by SSL for two-party communication. SGL integrates public key-based security into the communication and authorization capabilities using the Akenti server. In this peer-to-peer model, the PCCE server is no longer necessary. But, when it is running, it provides useful persistence information.

5 Conclusions

Although there has been significant progress in connecting collaborators for meetings, there is a need to address day-to-day interactions. Collaborators need mechanisms to track each other's progress and discuss current issues on an on-going basis. Tools that provide these capabilities are critical to providing the feeling that collaborators are working integrally together as a team as opposed to working on the same problem and only meeting occasionally. The Pervasive Collaborative Computing Environment provides critical capabilities required to enable collaborative computing because it addresses the base connectivity needed between users and provides a scalable solution to the collaboration environment. With accessibilty via the web, the PCCE also provides secure and easy-to-access interface.

References

[1] D. Agarwal, I. Foster and T. Strayer, "Standards-Based Software Infrastructure for Collaborative Environment and Distributed Computing Applications," white paper available at http://www-itg.lbl.gov/CIF/.

[2] D. Agarwal, ``Collaborating Across the Miles", Proceedings INMM/ESARDA Workshop on Science and Modern Technology for Safeguards, Albuquerque, NM, Sept. 1998.

[3] D.A. Agarwal, O. Chevassut, M.R. Thompson and G. Tsudik, "An Integrated Solution for Secure Group Communication in Wide-Area Networks," Proceedings IEEE Symposium on Computers and Communications, July 2001.

[4] D. A. Agarwal, S. R. Sachs, and W. E. Johnston, "The Reality of Collaboratories," Computer Physics Communications vol. 110, issue 1-3 (May 1998).

[5] K. Berket, D. A. Agarwal, P. M. Melliar-Smith and L. E. Moser. "Overview of the InterGroup Protocols." Proceedings International Conference on Computational Science, San Francisco, CA.

[6] P. Curtis, "Mudding: Social Phenomena in Text-Based Virtual Realities," Conference on Directions and Implications of Advanced Computing, sponsored by Computer Professionals for Social Responsibility, 1992.

[7] W. Edwards, "Policies and Roles in Collaborative Applications," Proceedings ACM Conference on Computer Supported Cooperative Work, 1996.

[8] I. Foster and C. Kesselman (eds.), The Grid: Blueprint for a New Computing Infrastructure, Morgan Kaufmann, 1999.

[9] J. Grudin, "Groupware and Social Dynamics: Eight Challenges for Developers." Communications of the ACM, Vol. 34(1), Jan. 1994.

[10] L. Jackson, "NCSA Habanero Java Object-Sharing Collaboration Framework," JavAus97 Proceedings, Queensland, Australia, Feb. 1997.

[11] K. Keating, J. Myers, J. Pelton, R. Bair, D. Wemmer, and P. Ellis, "Development and Use of a Virtual NMR Facility," Journal of Magnetic Resonance, December 1999.

[12] A. LaMarca, W. Edwards, P. Dourish, J. Lamping, et al, "Taking the work out of workflow: Mechanisms for document-centered collaboration," Proceedings European Conference on Computer Supported Cooperative Work, 1999.

[13] B. Nardi, S. Whittaker, E. Bradner, "Interaction and Outeraction: Instant Messaging in Action," Proceedings Conference on Computer Supported Cooperative Work, Philadelphia, PA (Dec. 2000).

[14] M. Perry and D. Agarwal, "Remote Control for Videoconferencing," Proceedings International Conference of the Information Resources Management Association, May 2000, Anchorage, Alaska. http://www-itg.lbl.gov/mbone/remote/

[15] M. Thompson, W. Johnston, S. Mudumbai, G. Hoo, et al., "Certificate-based Access Control for Widely Distributed Resources," Proceedings of the Usenix Security Symposium, Aug. `99.

[16] S. Whittaker, D. Frohlich, and W. Daly-Jones, "Informal Workplace Communication: What is it Like and How Might We Support It?" Proceedings of CHI'94 Conference on Human Factors in Computing Systems.

[17] N. Zaluzec, "Interactive Collaboratories for Microscopy & Microanalysis," Journal of Computer Assisted Microscopy, Oct. 1997.

[18] http://www.webdav.org/

[19] http://www.irc.org/

[20] http://www.aim.com/

[21] http://web.icq.com/

[22] http://vrvs.cern.ch/ or http://www.vrvs.org/

[23] http://www.webex.com/home/default.htm

[24] http://www-itg.lbl.gov/mbone/

[25] http://www-fp.mcs.anl.gov/fl/accessgrid/

[26] http://www.emsl.pnl.gov:2080/docs/collab/

[27] http://cvw.sourceforge.net/

[28] http://bscw.gmd.de/

[29] http://www.groove.net/

[30] http://www.csm.ornl.gov/enote/

[31] http://www-collab.ca.sandia.gov/

[32] http://www.itg.lbl.gov/BL7Collab/

[33] http://www.cs.wisc.edu/condor/