Support for Collaboration in Project Management

Dirk E. Mahling, Oddmar A. Sandvik,
and W. Bruce Croft

COINS Technical Report 90–16
March 1, 1990

Collaborative Systems Laboratory
Department of Computer and Information Science
University of Massachusetts
Amherst, Massachusetts 01003
Telephone: (413) 545 3147
Electronic Mail: mahling@, sandvik@, croft@cs.umass.edu

Abstract

Field studies reveal that project teams are not satisfied with current project management systems because only isolated activities are supported. As projects grow more complex and project teams grow larger, sophisticated tools and techniques are needed to support collaborative problem solving and communication tasks that are part of project management activities. In particular, representing activities, resources and objects explicitly allows a system to coordinate and facilitate group activities. The POLYMER system uses goal-based activity representations and planning techniques to support project description, exploration of alternatives, and monitoring. A project management interface, POLYMER-PM, employs project acquisition and display routines based on models of human task recall and project management processes. This interface maps project team demands to system functionality.

1 Introduction

Two perspectives dominate the current view of CSCW system design: one focuses on problem solving, the other on communication [CL84,KGE88]. These complementary perspectives must be integrated to effectively support complex cooperative tasks such as project management. Knowledge representation and planning techniques can offer part of this support [CRM80]. A formal representation of cooperation would help users to quickly explore alternatives, facilitate negotiation, detect scheduling and allocation conflicts, and allow them to cooperatively monitor the execution of their tasks.

The problems found in project management seem particularly appropriate for knowledge-based CSCW systems. Aspects of collaborative authoring, meeting support, and other CSCW domains can all be found in project management. In addition, project teams are used to providing the complex task specifications which are central to knowledge-based support.

During the initial stages of project management we can support communication to help people find common ground on the task and social levels [KGE88]. A goal-based project management system can help to formulate and share intentions and beliefs, collect and specify project knowledge from team members, allow joint development of project plans, and monitor progress through semi-structured messages.

The POLYMER-PM project management system, described in this paper, builds on earlier efforts in project management [BB89,AW86] and knowledge representation [SMR86], [CL88], by combining traditional project management tools with goal-based techniques and providing a consistent group interface. This combination allows users to capitalize on the benefits of knowledge representation and planning, while retaining familiar interfaces and functionality, such as Gantt charts or PERT diagrams. Advanced techniques, such as the reuse of existing project specifications, dealing with incomplete data, or delegating sub-projects, are needed to cope with large scale projects and project team involvement. Such functionality is an additional benefit to be gained from the goal-based representation of project knowledge.

We start by describing group activity in project management (section 2), then introduce a goal-based formalism to describe project knowledge (section 3), show how the implementation of the formalism results in techniques, data structures, and functionality to support cooperation in project teams (section 4), and finally present an interface, which facilitates cooperative project management (section 5). Section 5 also shows the use of interface and system in a project management scenario from the domain of organizing a conference. Implementation status and research perspectives are reported in section 6.

2 Project Management as Group Activity

Interviews with project managers in the construction and software development domains reveal that successful projects are contingent upon effective coordination of people and activities [PS90]. Coordination is in part achieved by communicating with project members, external groups, and other managers. Status meetings, progress reports, informal discussions, and client negotiation are all integral components of managerial tasks.

Four major kinds of communication were distinguished in the Pietras and Sandvik study [PS90]. Firstly, managers spend a significant amount of time coordinating project team members. Plans must be developed through negotiation, tasks assigned, and progress monitored. Similar communication takes place between the manager and external project dependencies: subcontractors, consultants, or other groups in the organization on which successful project completion is dependent. Thirdly, managers report progress and negotiate terms with higher level management. The final potential communication line extends to clients or client organizations, specifying deliverables and developing an understanding of client needs.

Coordinating a project group's individual efforts and facilitating collaborative planning and decision making are the primary responsibilities of a project manager. Team members are often involved in task assignments, plan specification, and plan refinement. Some individuals plan their own detailed subprojects and develop task duration estimates in cooperation with the manager. Resource assignment is very much a collaborative issue — the manager must consider the pool of available people, assess their skill levels and personal traits relative to the task at hand, and approach them with specific activities. Task breakdowns and duration estimates will be negotiated, and team members will thus be directly involved in plan development.

Monitoring project progress is also a collaborative effort. Group members attend status meetings and submit progress reports. This information must be assembled by the project manager and incorporated into the updated plan. Frequent formal and informal communications are means of detecting deviations from planned performance as early as possible. Managers tend to find plan updating a time-consuming and tedious task. This often deters them from using software tools. Schedule revisions must be distributed to team members and major changes require collaborative replanning.

Successful software development managers are found to expend much effort coordinating a common understanding of the application domain, system goals, and constraints among project team members [CKI88]. The manager should aid and direct the establishment of common representational conventions to build a shared system model and facilitate communication. The organization must also make efforts to mediate such models when intermediate work is transferred between project teams.

Subcontractors or other groups in the organization may constitute external dependencies in project plans. Small companies hire subcontractors or consultants to execute activities beyond their realm of expertise, while larger organizations focus on extensive efforts involving multiple projects and groups. Geographical distances, differing internal structures, and lack of shared plans and goals contribute to increased project complexity.

When dealing with external dependencies, the project manager is responsible for establishing channels of communication, negotiating project goals, obtaining commitments, and monitoring progress. Detailed project plans may be exchanged between groups, but typically the manager is concerned only with major milestones which are used to monitor the external group's progress.

The most common conflicts arising from such collaborative efforts are schedule slips.

The project manager is therefore concerned with detecting unmet milestones as early as possible, either by requiring plan updates at certain intervals or by sending observers to status meetings. The granularity of monitoring depends upon previous experience with the outside group, i.e. the level of credibility it has obtained. Upon detecting schedule conflicts, a period of negotiation ensues where schedules or functionality requirements may be cooperatively reformulated.

Project goals often originate from higher-level management. The process of goal formulation and negotiation tends to follow established channels of communication. From these objectives the project manager will produce a detailed plan.

Studies show that projects are viewed differently across varying levels in an organization [PS90]. Upper management is normally concerned with the major milestones or tasks, whereas a project manager must develop more complete plans. On the other hand, project team members may break their activities down to even higher resolution. Portions of the same plan thus holds varying attributes and contexts at different levels within the organization.

A project manager may face several challenges in client communication. Conflicting requirements, costs, schedules, and perhaps even plans must be negotiated and resolved during the planning stage. Deliverables and long-term plans must be carefully specified to avoid cost overruns or schedule slips. In such situations, the manager must engage in further negotiation with the client to develop an acceptable alternative plan.

Communication and collaboration are ill-defined components of managerial tasks, although they consume the majority of project managers' time. A tool must provide flexible cooperative support, but a knowledge-based approach can potentially aid the manager in coordinating activities and resources, as well as providing a means of reducing communication distances.

3 Representing Cooperative Activities

Planning and knowledge-based approaches are necessary to manage the increasing scope and complexity of projects [BB89,SMR86]. While traditional tools adequately fit small-scale tasks done by a single person, supporting cooperative work requires the coordination of many people and intelligent machines. Cooperative tasks are often "loosely structured"; there may be a *typical* way to accomplish certain milestones (goals) of a project, but the specifications are far from algorithmic. The representation language must capture any structure that is available, but must also be able to cope with flexibility and uncertainty. Yet the formalism should not be too complex to make it intractable for the system and incomprehensible to users.

A goal-based activity representation offers a powerful goal-subgoal matching mechanism, that allows specifications and views of projects on different levels of abstraction, thereby acknowledging the change in focus according to the levels of an organization and providing flexibility in achieving given subgoals. This flexibility enables the formalism to cope with changing environments and constraints. Preconditions allow the selection of activities that can achieve a subgoal in a certain situation. Experimental research, involving more than 180 subjects, shows that people can recall activities in a goal-based way [MC89] and that users can effectively employ advice generated from such a formalism [Mah90].

Cooperative activities are executed by a group of people or machines. The agent responsible for an activity in a project must be specified. To properly assign agents to tasks, a representation of the agent is required. This representation must include skills, experience, and current assignment. The descriptions of agents need not be confined to single people, but can include organizations as a whole, intelligent tools, and other resources, such as printers, processors, or vehicles. The formal description of cooperative activities, agents, resources, and objects involved, allows to generate plans appropriate in a specific situation to achieve milestones and jointly explore alternatives, while guaranteeing that constraints are not violated. This automatic guarantee of correctness allows users to concentrate on overall milestones, while the bookkeeping of details is performed automatically. Violations of constraints can easily be detected and flagged.

A tool designer using this formalism must consider the development of comprehensive domain and task models including the cognitive, social, and organizational processes found in project management activities. Broad communication and cooperation must be facilitated where tasks, people, and information are integrated to form an environment conducive to knowledge sharing and managerial control.

4 The POLYMER system

Over the past three years, the POLYMER system has been developed as a testbed for supporting cooperative work [CL88]. Using descriptions of project tasks and objects, POLYMER combines strategic and reactive planning [CRM80], negotiation, and explorations while interacting with multiple users to generate and explore plans for accomplishing a project.

4.1 System Overview

The overall architecture of the POLYMER system is presented in Figure 1. The large box shows POLYMER, the core planner, and its project descriptions. POLYMER uses activity descriptions to recursively expand milestones. This process results in plan networks (one for each project), which are stored and updated by a plan network maintenance system (PNMS). By recording the state of the planning process using an ATMS/Worlds system, the PNMS enables backtracking, constraint propagation, and truth-maintenance. People and machine resources, called *agents* and *objects* in the project descriptions, are assigned during the planning process. The project descriptions form a knowledge-base that contains four top-level types: relations, agents, activities, and objects. Agents include other POLYMER systems, people or groups, while objects subsume resources.

There can be a number of POLYMER systems at a given site or at different sites. A communication module makes distributed problem solving possible and facilitates communication between users and systems. In this way, knowledge can be shared, planners can exchange formal messages, and distributed task execution can be monitored.

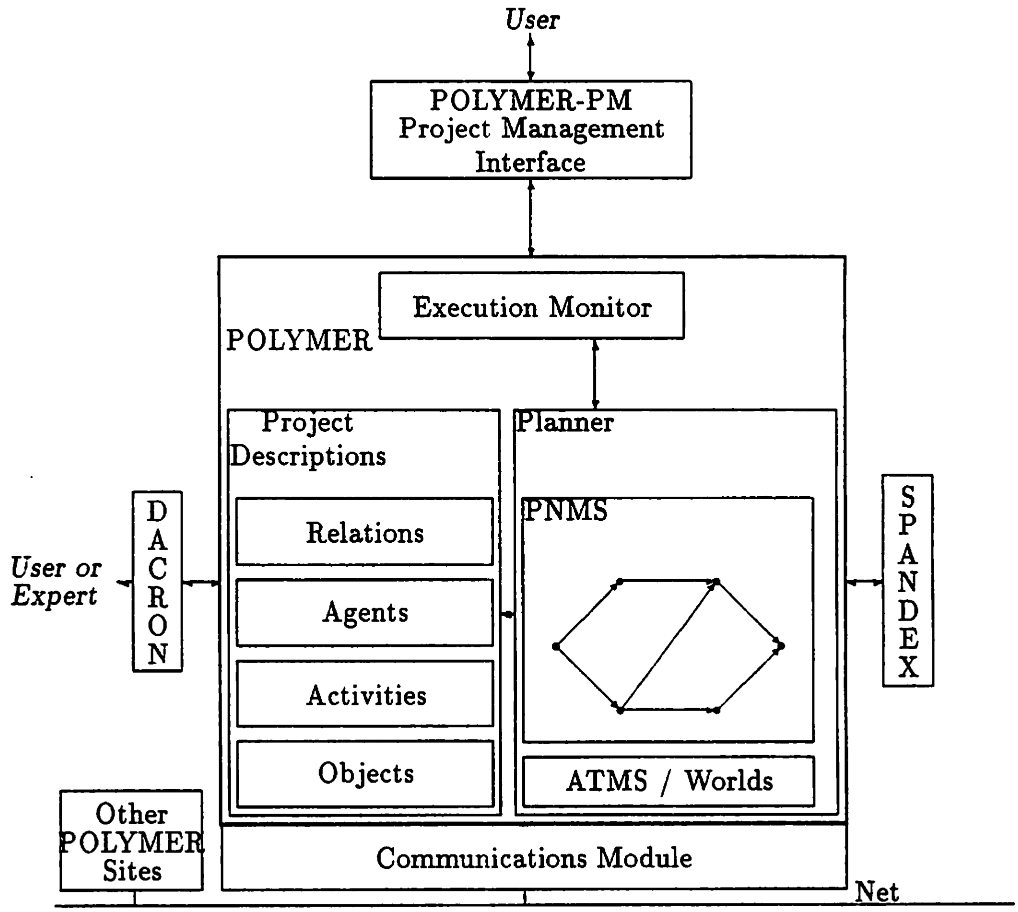

Figure 1: Distributed POLYMER Systems.

DACRON [MC88] allows the direct specification of knowledge by end users, who are often experts in their fields. It acts as a knowledge acquisition mediator between requirements of project descriptions and knowledge of end user. All four types of project descriptions can be acquired. SPANDEX [BC88] is an exception handler for POLYMER. When users execute tasks suggested by the system, the result might not exactly match the intended milestone. In that case, SPANDEX generates explanations for the mismatch and attempts to integrate the unexpected user action. The project management interface, POLYMER-PM, presents the general functionality of the POLYMER system, including knowledge acquisition and exception handling, to users in the project management domain. It is described in more detail in section 5, through a scenario showing collaborative system use.

4.2 Representing the Project Domain

POLYMER models the semantic context of a project through descriptions of *activities*, *objects* and *agents*. Using these project descriptions, POLYMER interactively generates

ACTIVITY: ORGANIZING A CONFERENCE

Milestone/Goal:

The conference has been organized successfully.

Preconditions:

Have a request from a Computer Society.

Served as co-chair on a previous conference.

Decomposition:

Prepare the conference.

Conduct the conference.

Compile final report.

Side-Effects:

Conference Proceedings are printed.

Plan Rationale:

Prepare conference, then conduct, then compile report.

Agents: Carol at ABC Corp.

Figure 2: A POLYMER Activity Description (with formal language paraphrased).

hierarchical, partially-ordered plans to accomplish goals shown as milestones in project management. The planning process, described below, is unique in applying a combination of "script based" and "goal directed" descriptions of activities to overcome the rigidity of scripts while greatly reducing the cost of purely goal driven systems. A sample activity description from the conference organization domain is shown in Figure 2. The representation includes both the *goal* and *preconditions* as well as a *decomposition* of the activity into smaller steps. The decomposition-steps are specified as lower level milestones for which other activities will be selected during the planning process. This non-procedural decomposition gives POLYMER the flexibility to achieve the same milestone in different ways, allowing it a situated response. On the lowest decomposition level the steps are tool invocations or commitments from other people. Causal and temporal relations between steps may be specified and used to generate orderings, as in PERT-charts. Planning and plan execution in POLYMER are interwoven in order to overcome the ambiguity and unpredictability in areas such as project management. A strategic plan is pursued until problems arise during execution. POLYMER then switches to tactical re-planning techniques to navigate the impasse.

To help managers develop a first draft of a project outline, POLYMER attempts to generate an almost complete plan to accomplish the given milestone. The plan is constructed in a hierarchical manner, using the formal description of the milestone and current constraints of the world. This helps ensure the situated relevance of the plan. The planning proceeds top-down and left-to-right. The users can direct which branch of the project network to develop next. The user can also explore alternatives by letting the planner backtrack, and try different scenarios. During execution, this strategic plan is updated to react to changes in the environment.

5 A Collaborative Project Management Interface

Studies show that managers tend to work on multiple projects or aspects of projects simultaneously [PS90]. This has prompted us to extend *rooms metaphor* [Mah85,Cha84,CH87] as a foundation from which to develop an interface for collaborative project management.

Users inhabit their own *office suites*, each of which is comprised of a *lobby* and a number of user-defined *rooms*. Every such room is defined by a particular screen layout of windows and tools, and is created and furnished by the user. The lobby is the front room of each suite, providing facilities for creating or entering rooms, help and information, and other tools. Entering a *door* signifies a change in work context and back doors permit direct access between rooms. Such rooms can be shared between people and protection facilities therefore enforce strict user control of information access.

Our scenario for illustrating the use of this interface is selected from the domain and of planning conferences, and focuses on some of the advanced project management functionality POLYMER-PM can provide, rather than on basic features. All agents within the given organization are therefore assumed to inhabit a POLYMER-PM site. The people in this scenario are Carol, a manager, Tracy, her secretary, and Bob, another manager in the organization.

5.1 Starting a New Project

This section will illustrate how to create and specify a new project in a dedicated room.

Carol receives a request from IEEE to organize a conference on visual languages. The request includes major constraints: the 3 day conference should take place in the US during October 1993, and approximately 1,000 people are expected to attend. She accepts the request and immediately begins organizing the conference.

She dedicates a room to the conference project and creates a new door in the lobby titled `IEEE-VL93`. She enters the room and furnishes it by selecting project file cabinets and the mail and calendar tools from a menu. Carol opens the `NEW` file cabinet and creates a folder titled `IEEE-VL93` to initiate a new project. She consequently begins to develop a project plan by creating an `Organize Conference` icon in the tree-structured Work Breakdown Structure (WBS) tool. Using direct manipulation techniques, she breaks this goal into three high-level milestones: `Prepare Conference`, `Conduct Conference`, and `Final Report` (see Figure 3). We will focus on the conference preparation task.

5.2 Finding Alternative Plans and Accessing Shared Knowledge

This example describes how POLYMER-PM can aid in finding alternative plans to achieve a milestone, reusing previous project descriptions, accessing shared organizational knowledge, and adding temporal constraints.

Carol now specifies the formal goal of the `Prepare Conference` activity: *The status of the conference is prepared.* Since she helped prepare an ACM conference two years ago, a plan to achieve this goal is already stored in her project knowledge base. POLYMER-PM

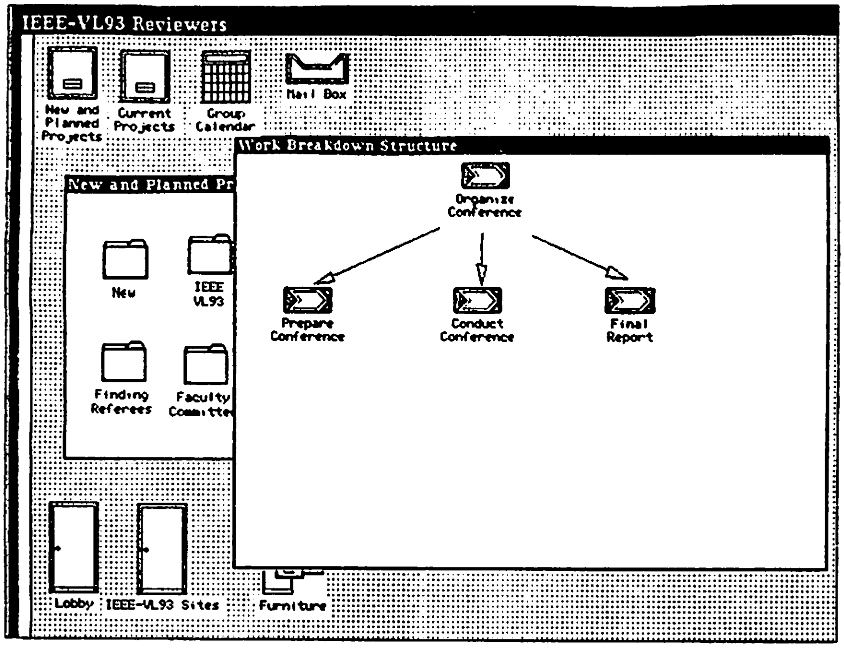

Figure 3: Initial Work Breakdown of Conference Plan

is thus able to make use of existing knowledge and construct a plan for the milestone. The suggested project plan is displayed in an activity editor based on a model of human task recall [MC89].

Carol inspects her previous ACM conference outline, and decides that it is not appropriate for IEEE-VL93. She queries the organizational knowledge base through POLYMER-PM for other plans that can achieve this milestone. In our example, Bob, another manager, was once responsible for planning a previous IEEE conference. He has provided this project information in the shared knowledge base. The two alternative plans are shown in Figure 4. Carol selects the IEEE plan, which is inserted into the WBS window as a hierarchy of tasks and milestones.

Carol decides to add temporal constraints to this partial plan: `Prepare Conference` must be completed by June 1993, and `Conduct Conference` has a duration of 3 days and takes place during October 1993. She also decides that `Receive Submissions` must be completed by April 1, 1993 and `Referee Submissions` has a duration of one month. These constraints are automatically assigned to subprojects. A more complete plan is shown in Figure 5.

5.3 Delegating Tasks and Sharing Rooms

Carol delegates project activities to other people in the organization. A shared room is used to facilitate collaboration between these agents and allow joint planning. POLYMER's subgoal mechanism supports the concurrent development and execution of parts of a larger plan.

Since Bob has previous experience in selecting referees, Carol asks him to take over this

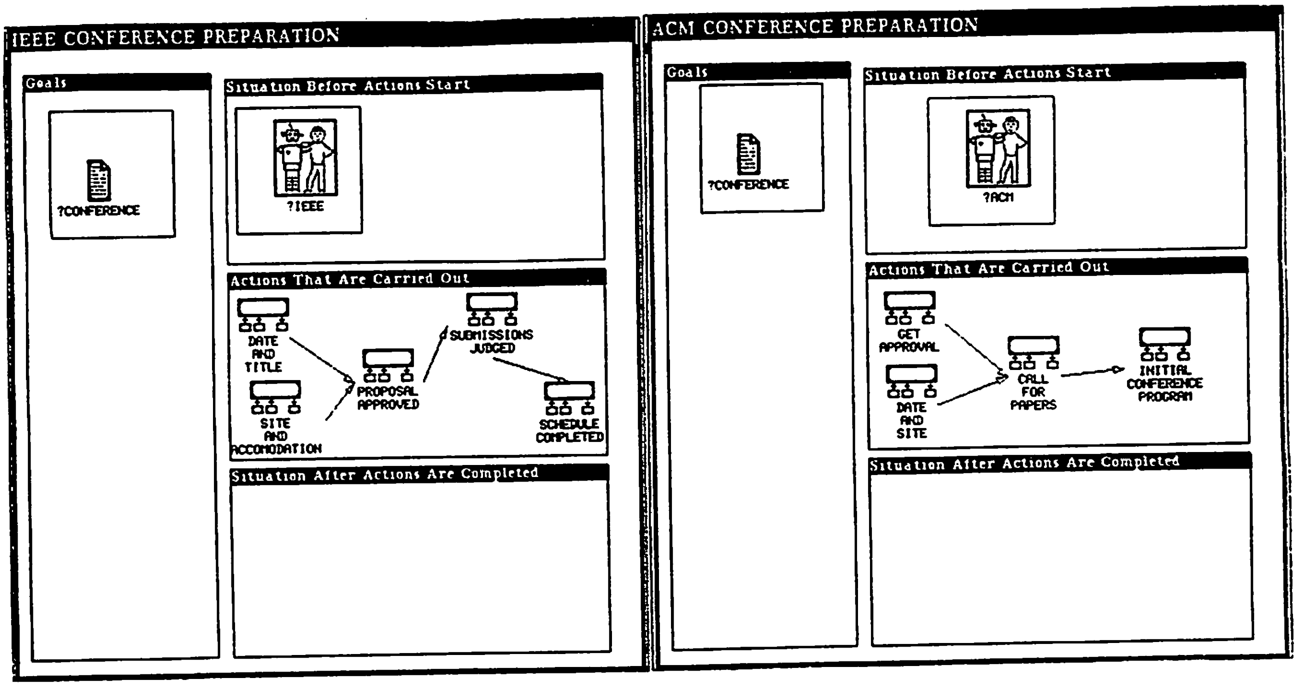

Figure 4: Alternative Plans to Achieve a Milestone.

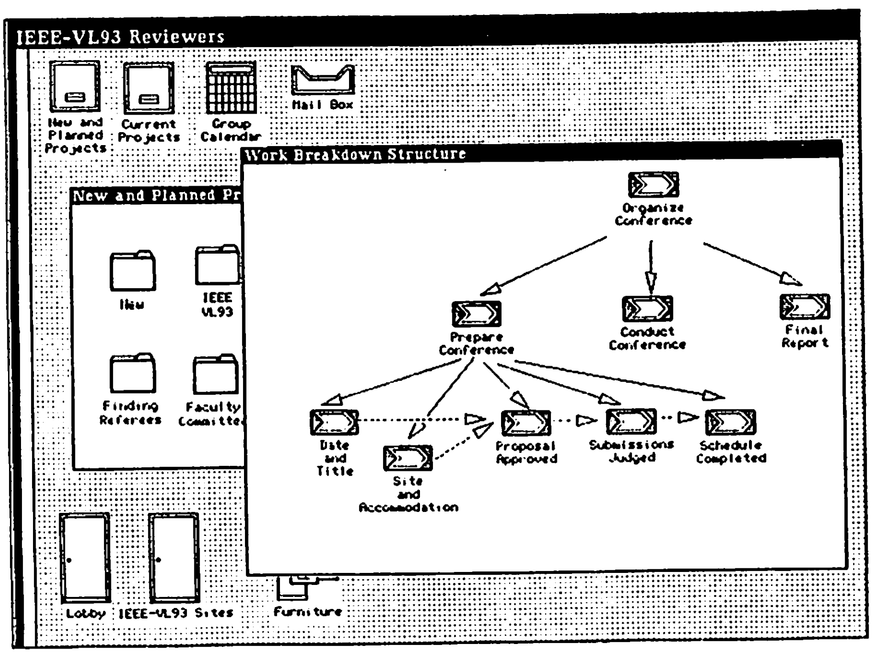

Figure 5: Work Breakdown of Conference Plan.

task. In order to draw on available resources, Carol requests Tracy, her secretary, to initiate a long-term collaboration and discussion forum. They decide to share a room in Tracy's suite and agree upon a password (or key) for this room, which will be used to cooperatively plan site and accommodation selections. In this room, they will elaborate and update the plan as well as post notes to each other (see Figure 6). This forms a collaborative work area which can be flexibly shared and laid out by the cooperating agents.

In this particular scenario, many planning tasks will be carried out by the manager rather than by team members. However, POLYMER-PM knows from Carol's resource library that Tracy has a history of sending out proposals, distributing calls for papers, and receiving submissions. Upon request, these tasks are allocated to the secretary.

5.4 Plan Exploration, Negotiation, and Updating Knowledge

In this section, project members will use POLYMER-PM to explore alternative plans and communicate task completion. Negotiation with other agents takes place, and resource knowledge is used and updated.

To determine the conference site, Tracy draws upon her own resource library to get suggestions on appropriate locations. Given the situation's constraints and previous knowledge of sites, POLYMER-PM can be used to compile a list of potential locations that fit these criteria. Tracy leaves this list, annotated with her own recommendation notes, in the shared room, shown in Figure 6. Carol then chooses a small set of site alternatives and issues requests. The final decision is taken after conferring with IEEE by electronic mail.

POLYMER-PM can also be used to communicate plan updates between sites. After receiving all submissions and entering them in the resource library under subject areas, Tracy notifies the tool about the completed task. This appears in Bob's plan and he sets out to obtain a set of potential referees, searching by the papers' subject areas.

The next step is to match referees with papers. The project manager will need POLYMER-PM's backtracking facility to explore various resource allocation alternatives for an optimal match. Bob defines semi-structured referee request templates to be filled out by the user as well as the tool and distributed via regular electronic mail. Some requests are turned down, and backup candidates are notified. Others cannot complete the task within the deadline, and the manager must engage in replanning, and negotiating alternative solutions with the people in question. Finally, some replies indicate that the referee did not have the specific expertise required. Bob uses this information to update his resource library for future use and explores alternative solutions. Schedule changes are sent to referees and the IEEE using predefined templates. POLYMER-PM thus aids in reducing the communication overhead by using knowledge about project management collaboration, retains flexibility and ease in supporting human communication, and integrates with existing electronic mail systems.

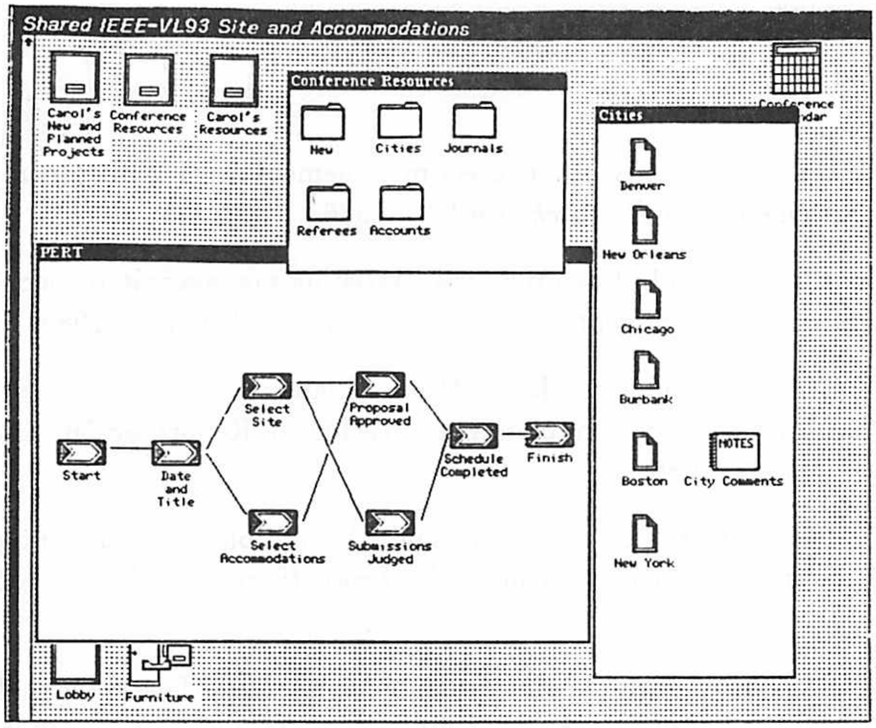

Figure 6: Room Shared Between Collaborating People.

6 Status and Future Work

The underlying POLYMER planning and representation system has been implemented and tested. The knowledge acquisition module DACRON and the exception handler SPANDEX are also completely implemented. DACRON, based on a model of human task recall, serves as a conceptual basis for our project management efforts. The knowledge display routines in DACRON, which have been extensively tested in usability engineering experiments with more than thirty subjects, constitute display and acquisition techniques for project management. The POLYMER-PM interface, tailored to the idiosyncrasies of project management, is under current development. Studies of human planning and project management accompany this development, as do user studies with rapid prototypes. A generic theory of collaboration [MCC90] that is being continuously refined guides the design of the distributed version of the POLYMER, called D-POLYMER.

7 Acknowledgments

We would like to thank Christine Pietras of the Industrial Engineering Department, for collaborating in researching the project management domain. Thanks also to Wendie Wulff of FRS-EDL and the members of the Collaborative Systems Laboratory for providing constructive feedback on earlier versions of this paper.

References

[AW86] A. Assad and E. Wasil. Project management using a microcomputer. *Computer and Operations Research*, 13(2/3), 1986.

[BB89] K. Bimson and L. Boehm Burris. Assisting managers in project definition: Foundations for intelligent decisions support. *IEEE Expert*, 1989.

[BC88] C. A. Broverman and B. C. Croft. Plausible explanation to cope with unanticipated behavior in planning. Technical Report 88-56, University of Massachusetts, 1988.

[CH87] S. K. Card and D. A. Henderson. A multiple, virtual-workspace interface to support user task switching. In *Proceedings of CHI+GI 1987*, pages 53–60, 1987.

[Cha84] P. P. Chan. Learning considerations in user interface design: The room model. Technical Report CS-84-16, University of Waterloo, Computer Science Department, 1984.

[CKI88] B. Curtis, H. Krasner, and N. Iscoe. A field study of the software design process for large systems. *Communications of the ACM*, 31(11), 1988.

[CL84] B. Croft and L. Lefkowitz. Task support in an office system. *ACM Transaction on Office Systems*, 2:197–212, 1984.

[CL88] W. B. Croft and L. S. Lefkowitz. Knowledge-based support of cooperative activities. In A. H. Bond and L. Gasser, editors, *Readings in Distributed Artificial Intelligence*, pages 599 –605. Morgan Kaufman, 1988.

[CRM80] E. Charniak, C. Riesbeck, and D. McDermott. *Artificial Intelligence Programming*. Lawrence Erlbaum, Hillsdale, 1980.

[KGE88] R. Kraut, J. Galegher, and C. Edigo. Relationships and tasks in scientific research collaborations. *Human Computer Interaction*, 3(1), 1988.

[Mah85] D. E. Mahling. Beschreibung und Bewertung von Fenstertechniken in der Dialoggestaltung des Apple Macintosh und des Siemens PC-X nach kognitive-ergonomischen Kriterien. (evaluation of apple macintosh and siemens-pc windowing techniques according to cognitive-ergonomic criteria). Master's thesis, Brunswick Institute of Technology, Department of Psychology,Germany, 1985.

[Mah90] D. E. Mahling. *A Visual Language for Knowledge Acquisition, Display and Animation by Domain Experts and Novices*. PhD thesis, University of Massachusetts, February 1990.

[MC88] D. E. Mahling and W. B. Croft. Knowledge acquisition for planners. In B. Gaines and J. Boose, editors, *Proceedings of the 3rd Knowledge Acquisition for Knowledge Based Systems Workshop*, 1988.

[MC89] D. E. Mahling and W. B. Croft. Relating human knowledge to plans. *International Journal of Man-Machine Studies*, 31:61 – 97, 1989.

[MCC90] D.E. Mahling, B.G. Coury, and W.B. Croft. User models in cooperative task-oriented environments. In *Proceedings of the 23rd Hawaii International Conference on Systems Science*. IEEE, 1990.

[PS90] C. Pietras and O. Sandvik. A review of project management tasks and tools. Technical Report CSL-90-1, University of Massachusetts, Collaborative Systems Laboratory, 1990.

[SMR86] A. Sathi, T. E. Morton, and S. F. Roth. Callisto: An intelligent project management system. *AI Magazin*, Winter, 1986.