Flexible, Active Support for Collaboration with ConversationBuilder

Simon M. Kaplan, William J. Tolone, Douglas P. Bogia and Theodore Phelps

Human-Computer Interaction Laboratory
Department of Computer Science
University of Illinois at Urbana-Champaign
Urbana, IL 61801, USA
+1 217 244 0392 {kaplan,tolone,bogia,phelps}@cs.uiuc.edu

ABSTRACT
We overview the ConversationBuilder system and its demonstration at INTERCHI 93.

KEYWORDS
Collaboration Environment

ABOUT THIS DEMONSTRATION
ConversationBuilder [1] (CB) is an *open architecture for collaborative work* which builds on earlier efforts [2] [3]. Our goal has been to create an environment that provides active support to its user community in an open and flexible way. Open support means the tools can be tailored to different domains and cultures of collaborative activity. Active support means that users are provided context-sensitive help by the system, enabling them to find out how they got into a situation, what can be done now, what others are doing, and what obligations they have. Active support is provided through our ability to define protocols to the system, which in turn utilize the mechanisms provided by the CB kernel.

Flexible support means that the system can dynamically alter to reflect changed situations. For example, if a user is waiting on approval of a project from a manager, who goes on vacation, the task dependency structure in the system can be modified to remove the approval, pass it to another, and so forth. Further, as one works on a task one tends to find problems that should be fixed by others, and so forth. The ability to work on multiple tasks simultaneously, context switching among them, and evolve arbitrary webs of obligations, allows us to model this kind of behavior on the fly.

CB also has a notion of shared hypertext, with hyperlinks. Hypertext nodes can be used to hold the shared information manipulated by users, such as program text, documents, IBIS issues, positions and argu-

ments, and so forth. The types of nodes and links in the system are defined as part of protocol specifications.

CB supports a notion of user awareness: the system has WYSIWIS features, and this is used to keep users aware of each others actions.

As part of the development of CB we've developed a set of interesting new tools, including a message bus for tool interconnection, a widget server for dynamic tool interface construction and sharing, graph browsers and shared drawing tools. We are investigating techniques and mechanisms for the dynamic, collaborative construction of dynamic user interfaces by multiple applications; the demonstration will show off the entire CB system through use of these facilities.

We have deployed CB in several different application domains by writing appropriate protocols. CB has been used to support synchronous collaboration in situation such as semi-structured design activities of the IBIS or Design Rationale type, and code walkthrough/review meetings. Examples of asynchronous support include the modeling of a software quality assurance process, and the modeling of a shared versioning and configuration management system, which is used to maintain complex documents such as the source code and documentation of the CB system itelf.

REFERENCES
[1] Simon M. Kaplan *et al.* Flexible, active support for collaborative work with conversationbuilder. In *Proc CSCW 92*, November 1992.

[2] Fernando Flores *et al* Computer Systems and the Design of Organizational Interaction. *ACM TOIS*. 6(2), April 1988.

[3] Allan Shepherd *et al.* Strudel: an extensible electronic conversation toolkit. In *Proc CSCW 90*, October 1990.