

10-2007

# Statistics and Analysis Tools for a Computer-Supported Collaborative Learning System

L. D. Miller

lmille@cse.unl.edu

Adam Eck

aeck@cse.unl.edu

Leen-Kiat Soh

University of Nebraska, lsoh2@unl.edu

Hong Jiang

University of Nebraska - Lincoln, jiang@cse.unl.edu

Follow this and additional works at: <http://digitalcommons.unl.edu/cseconfwork>



Part of the [Computer Sciences Commons](#)

---

Miller, L. D.; Eck, Adam; Soh, Leen-Kiat; and Jiang, Hong, "Statistics and Analysis Tools for a Computer-Supported Collaborative Learning System" (2007). *CSE Conference and Workshop Papers*. 18.

<http://digitalcommons.unl.edu/cseconfwork/18>

This Article is brought to you for free and open access by the Computer Science and Engineering, Department of at DigitalCommons@University of Nebraska - Lincoln. It has been accepted for inclusion in CSE Conference and Workshop Papers by an authorized administrator of DigitalCommons@University of Nebraska - Lincoln.

# Statistics and Analysis Tools for a Computer-Supported Collaborative Learning System

L.D. Miller, Adam Eck, Leen-Kiat Soh, Hong Jiang

University of Nebraska-Lincoln, Computer Science & Engineering,  
Lincoln, NE 68588 {lmille, aeck, lksoh, jiang}@cse.unl.edu

**Abstract** - Traditional computer-supported collaborative learning (CSCL) systems focus primarily on facilitating group work amongst students, providing several different modes for different types of communication, from chatting applications to shared whiteboards. The power of a good CSCL system, in addition to the interactions it supports, lies with the data the system can gather from those interactions (both relational and content-based). Thus, we have developed the Intelligent Multiagent Infrastructure for Distributed System in Education (I-MINDS), a CSCL system built on the Microsoft ConferenceXP platform that tracks real-time data to support student learning and classroom management. I-MINDS facilitates (1) student-student interactions through message dialogs, stored as trees, and (2) student-teacher interactions through question/answer pairs. I-MINDS logs these interactions in a MySQL database. Our VirtualClassroom analysis tool derives statistics from these interactions such as student responsiveness, degree of dialog involvement, amount of contribution, and quality of involvement and contribution. Then, an instructor can view overall trends for individual students and groups and detailed charts for all statistics for each student or all students for each statistic, which is especially helpful in large classrooms and in distance education. The instructor can also adjust a “highlight” threshold for each statistic such that any value failing to meet the threshold will be highlighted for easy observation. Our DatabaseSearch analysis tool allows students and teachers to search through recorded interactions and filter the results based on metadata such as time interval, class relevance, or keyword content.

*Index Terms* – Computer-supported collaborative learning, Statistics and Analysis tools, Virtual classroom.

## INTRODUCTION

Currently, one of the main joint projects on the frontiers of computer science and education research is determining how to devise and deploy software tools to support classroom instruction. One popular approach is computer-supported collaborative learning (CSCL), a type of system that advocates group collaboration to promote learning amongst students. With this software, students have multiple modes with which to work in tandem with their peers to solve group problems or

tutor one another, including instant messaging and shared whiteboards.

While it is important to provide means of communication, these CSCL systems can also monitor student performance and provide intelligent support to promote learning and classroom management. By doing so, the systems can include the ability to pinpoint the weaknesses of individual students as areas for the teacher to target in her instruction, as well as determine the strengths of the students in order to assign group roles and membership. For example, by automatically matching strong students with weaker ones, the students can learn from each other to advance both of their educations. Thus, it is equally important for CSCL systems to be able to provide insightful analysis of students to support teachers and students alike in their decision making process in the classroom.

While designing a CSCL application built to run within the ConferenceXP (CXP) collaborative platform [1] developed by Microsoft Research, based on the Intelligent Multiagent Infrastructure for Distributed Systems in Education (I-MINDS) [2], we needed a set of principles to guide our creation of the analysis tools required to provide classroom support. Using the ideals behind collaboration management [3] as a strong influence, we came up with the following five principles, which we feel are important for the analysis tools within intelligent CSCL systems to follow: (1) Generate both content and relational information from tracked user interactions, (2) Display generated information (e.g., statistics) in an organized, informative manner, (3) Allow for user customization, (4) Offer intelligent processing of data for automated support, and (5) Provide both asynchronous and synchronous operations.

Based on these principles, we created several tools for assisting both instructors and students in a collaborative environment. For teachers, we developed VirtualClassroom, an analysis tool for classroom management that provides instructors with a quick overview of the current state of their classrooms by displaying the content of recent student interactions, as well as statistical representations of each student, which highlight their strengths and weakness in graphical form. Finally, because our system tracks all data sent between users, we also provide a search engine called DatabaseSearch for querying archived content based on tracked metadata, such as sender, message content, and score. This tool allows users to analyze previous interactions and find relevant information to the tasks at hand.

In the following, Section 2 provides some background and related work on the use of analysis tools in CSCL system, as well as a brief introduction to both ConferenceXP and I-MINDS. Section 3 provides an in-depth description of our design principles, while Section 4 describes how we implemented them into CXP+I-MINDS and its analysis tools. In Section 5 we describe two deployments of CXP+I-MINDS, and we finish up in Section 6 with some conclusions and a discussion on possible future work.

## BACKGROUND AND RELATED WORK

In this section, we provide some background and related work on the design of the analysis tools used in the CXP+I-MINDS computer-supported collaborative learning (CSCL) system. We begin by describing the process of collaboration management, from which we developed our five principles of analysis tool design. Secondly, we provide some background on both the ConferenceXP collaborative framework and the Intelligent Multiagent Infrastructure for Distributed Systems in Education (I-MINDS). We finish the section with a review of analysis tools used in several other CSCL systems.

### *I. Collaboration Management*

In order to provide a paradigm upon which to improve intelligent support within CSCL systems, Jermann, Soller, and Muehlenbrock [3] developed the concept of collaboration management, a four-step process influenced by the work of Barros and Verdejo [4]. This process describes the functionalities a metacognitive system must provide to enable the system to coach users in order to guide and improve their performance. These four steps include: (1) Tracking user data, (2) Processing user data into indicators of performance, (3) Diagnosing collaboration based on the indicators, and (4) Choosing an appropriate action based on the diagnosis.

While this paradigm is useful for automated, intelligent support, a system only needs to provide the first two steps, while the last two can be performed by a student in order to improve her own collaboration, or a teacher to support her classroom [4]. This latter option is possible if the indicators are displayed in some useful form (most often graphically) to users that will analyze and act upon that information. We adopted the above paradigm when devising the five design principles outlined in Introduction.

### *II. ConferenceXP Framework*

The ConferenceXP collaborative framework [1] is an open project developed by Microsoft Research to support online collaboration in educational and industrial settings. This system provides multiple means of communication for users, including text-based chatting, PowerPoint presentation support, shared whiteboards, and audio/video streaming. Because ConferenceXP is an open project released under the Shared Source license, researchers are free to develop their own applications to plug-in to the CXP framework and benefit from its multimedia and networking capabilities without having to worry about the underlying implementation details.

### *III. I-MINDS*

The Intelligent Multiagent Infrastructure for Distributed Systems in Education [2] is a framework for intelligently supported student collaboration and classroom management through the use of intelligent personal agents. Every user is given her own software agent which monitors student activities to provide personalized support to their user. This support can range from alerting the user of personal conditions, such as low contribution levels or good motivation, to promoting collaboration through assisting the user in selecting collaborative partners based on its interactions with other user agents.

I-MINDS also provides two other types of agents: group agents and teacher agents. The group agent is in charge of modeling group interactions for analyzing group performance, as well as helping restructure the groups when problems arise. Teacher agents are responsible for insuring that all students receive the proper content from the instructor, as well as organizing classroom information to present to the teacher to help with classroom management. Additionally, teacher agents can assist overburdened instructors by automatically answering students' questions based on whether or not a new question is considered relevant or equivalent to previously answered questions.

### *IV. Other CSCL Systems with Analysis Tools*

One example of the use of analysis tools in CSCL is the Collaboration Analysis Toolkit (CoAT) [5], which was developed for the ModellingSpace environment. Its purpose is the analysis of users' actions during collaborative problem solving activities. CoAT was designed to read audio and video streams in addition to log files stored in XML format. These log files store user actions (events) according to the timestamp and the user ID. The analysis tool allows the researcher to asynchronously view the relationships between events, organized by task or goal levels. CoAT also provides the means to reenact the entire log file, enabling the user to construct a multi-level annotation scheme. The interface for CoAT allows the user to move events from multiple sources, offering user customization of both the presentation and underlying information. However, the CoAT analysis tools do not track the relations between user interactions, allow for synchronous analysis, or offer intelligent data processing in its analysis tools.

Another analysis tool found in CSCL systems is the Collaborative Interaction Analysis tool [6], a part of the Synergo collaborative environment. This analysis tool consists of several views where statistics are calculated and presented to the user in the form of charts and tables based on data saved in log files. The user can also customize the statistics presented in the views by providing metadata such as the time intervals for specific events. The analysis tool is capable of operating synchronously, allowing teachers to track how students are working and provide comments and suggestions as feedback. In an asynchronous environment, the analysis tool can be used to provide step by step playback of an activity. However, the analysis tools for Synergo lack the ability to store relational

information between events, as well as intelligent data processing (see subsection IV in the next section).

A third CSCL system reviewed is DEGREE [4], which consists of four levels, one of which is the analysis level. In this level, user interactions are analyzed both quantitatively and qualitatively. DEGREE uses a fixed conversational structure where user interactions are defined as one of six categories and stored in relation to other messages through a tree-like structure. The analysis level is designed to operate synchronously, allowing an advisor module to provide feedback to users. All the user interactions are stored in a relational database, which facilitates asynchronous operation, as well. The analysis tools also include numerous options for presenting the statistics graphically for all users. These tools also process the data through various knowledge bases and intelligently provide responses to users, but they are not customizable by users.

CXP+I-MINDS, on the other hand, adheres to all five design principles to be discussed in the next section. Details of this comparison are presented later in the Conclusions section, after we describe the design and use of the analysis tools within CXP+I-MINDS.

## DESIGN PRINCIPLES

In this section, we describe the design principles presented in Introduction in further detail. Note also that, in the following discussions, a *session* is an arbitrary interval during which the CSCL system is operating and able to record user data.

### I. Generate Both Relational and Content Information from Tracked User Interactions

There are two important forms of information that can be derived from data tracked during user interactions: *relations* and *content*. The content of data is the meaning as a direct result of a user's action. For example, if a user sends a message (e.g., "I worked on that calculus problem for three hours last night"), the content is the idea represented by the words comprising the message, and can later be used to generate higher orders of information, such as student interests and knowledge (e.g., the student knows about calculus and is motivated to solve calculus problems).

Relational information represents how some piece of data relates to other data. Using the same example as above, the message sent by the student has three forms of relational data. First of all, the message has relation to *time*: it was sent in a specific time interval and during a specific session. It also has relation to *people*: the message was sent by a specific person to a specific group of people. Finally, the message also has relation to *other messages*: it belongs to a specific dialog which might include other messages, including a response from the student's group members. This relational data is useful for indicators such as determining whether or not actions by users are on topic, how quickly a user responds to other users' actions, and how active a specific user is in their collaborative group.

By combining both relational and content information from user interactions, different indicators of student and

group performance can be computed and used to diagnose collaboration, as well as provide support in case of problems. This information also provides multiple ways of searching for archived data, which is useful for analyzing prior interactions.

### II. Displaying Generated Information in an Organized and Informative Manner

For the second design principle, the interface with the analysis tools should be able to present the data in a visually convenient manner in real-time. To meet the first aspect of this design principle, the analysis tools should be able to generate visual representations of the information such as charts and figures. This design principle is especially important for analysis tools operating in a synchronous environment. Presenting the teacher with a chart consisting of the average number of student responses about the material presented is far more useful than presenting the raw statistics when the teacher is trying to gauge student understanding during the current session.

To meet the second aspect of this design principle the visual representation must be available in real-time. The way data is stored by the CSCL system has huge impact on meeting the second half of this design principle. Distributed or local storage of the data can potentially introduce delay into computing the statistics and result in consistency issues about the data. The use of a relational database to store the statistical data is advisable. This allows statistics consisting of complex queries to be executed quickly by the database and reduces the consistency maintenance problems.

### III. Allowing for User Customization

The third design principle states that the analysis tools should be customizable. This does not necessarily refer to the way statistics are presented in the interface. Users should certainly be able to decide which statistics are important and which should not be displayed in the interface. However, the purpose of analysis tool customization is to allow the user to introduce domain knowledge for meta-analysis of the statistics collected by the CSCL system. We propose that this be done by allowing the user to apply filters and thresholds for the predefined statistics. This is particularly useful for the teacher allowing her to determine when the statistics for her students fail to meet her expectations.

### IV. Offer Intelligent Data Processing for Intelligent Support

For the fourth design principle, the analysis tools should be capable of intelligently processing the data. For analysis tools to meet this design principle, they must have the capacity to process the data without direct user involvement. However, this is not to say that the analysis tools must ignore user involvement. Instead, the analysis tools must go beyond simply handling user requests. Examples of intelligent data processing include automatic sorting of data, resizing of charts, and prompting the user when a statistic exceeds a threshold. Analysis tools which make use of artificial intelligence (AI) such as natural language processing and machine learning

certainly meet this design principle, as do those that utilize knowledge-bases.

#### V. Provide both Synchronous and Asynchronous Operations

The final design principle states that the analysis tools should be capable of operating in both synchronous and asynchronous environments, sometimes referred to as action-oriented and text-production oriented [7]. This implies that the statistics will be available to users during a session and in subsequent sessions, as well as during offline operation. Analysis tools operating asynchronously support longitudinal evaluation of the statistics collected from the CSCL system. This is a powerful tool for teachers who want to track how their students are doing over the course of the semester. Additionally, the analysis tools could be used asynchronously by students to review dialogs or even replay an entire session. This is certainly useful for students who want to review a missed section.

### IMPLEMENTATION

CXP+I-MINDS was built using the C# programming language on top of the ConferenceXP platform developed by Microsoft Research. ConferenceXP [1] handles the entire networking layer for our CSCL including both sending and receiving data from user interactions. Presently, though there are multiple modes of interaction in I-MINDS (audio, video, whiteboard, etc.), our statistical analysis tools focus on deriving statistics from messages communicated among the users. The messages in I-MINDS are stored in a tree-like structure called a message tree; the message which started the dialog is stored at the root, responses to this message are stored as leaves which can then be responded to, causing the tree to grow.

The backend of the CXP+I-MINDS consists of a relational database using MySQL 5.0. All the user interactions tracked by the system are logged here. This database consists of over 100 fields in 20 tables. Due to page restrictions, we do not include the database design here. The heart of the database is the *Messages* table where user interactions are stored with links to the session, classroom and participants involved.

CXP+I-MINDS also has a logger module that runs automatically when there is an active session. The logger simply monitors message traffic between human users and logs the traffic to the database. Each classroom has only one logger in an effort to reduce the number of necessary database connections, as will be discussed in the deployment section later.

In the following, we first discuss the statistics and measurements derived by CXP+I-MINDS and how they are grouped into indicators. Next, we present two primary applications within I-MINDS: BuddyGroup and QuestionAnswering. Then, we describe two analysis tools built to process and display the information derived from the data collected by the applications: VirtualClassroom and DatabaseSearch.

#### I. Statistics and Measurements Tracked by I-MINDS

I-MINDS currently tracks seven measurements and seven statistics based on the data collected from user interactions. These measurements and statistics are grouped into five dif-

ferent indicators: *Buddy Messages*, *Teacher Messages*, *Topics Started*, *Responsiveness*, and *Dialog Involvement*.

The *Buddy Messages* indicator consists of one measurement of the number of messages sent to the user's buddy group in the current session and a statistic for the average of this measurement over all sessions. The *Teacher Messages* indicator contains the same measurement and statistic for messages sent to the teacher. Both these indicators represent how active the user is during classroom activities. The *Topics Started* indicator is comprised of one measurement for the number of dialogs started by the user and statistics which calculate the average number and quality of the resulting messages. This indicator provides information about the quality of dialogs initiated by the user. The *Responsiveness* indicator contains measurements for the number of messages sent and received by the user and a statistic for the user's average response time. The *Dialog Involvement* indicator is composed of measurements for the responses sent by the user in dialogs the user initiated and dialogs initiated by others. It also contains statistics for the average number of responses sent for both types of dialogs. This final indicator can be used to provide a model for how well the user collaborates over different dialogs started by different users.

#### II. I-MINDS Applications

I-MINDS has two primary applications: BuddyGroup (BG) and QuestionAnswering (QA). Each has different functionalities for teachers and students. For the BG application, in the student version, only messages within dialogs started within a collaborative group are displayed, but in the teacher version all messages sent are automatically displayed. This allows the teacher to keep track of the dialogs students are involved with. For the QA application, the student version only allows students to ask questions and view the responses. The teacher version displays all the pending questions and those that have been answered or discarded previously in a list form.

The BG application facilitates student-student interactions through the use of collaborative groups, or "buddy groups". For each user, these collaborative groups consist of a subset of the students using the CSCL system. Messages sent during the current session by students using the BG application will only be visible to their buddies. The buddy relationship is symmetric, meaning that both students must agree to be buddies to form a group. Also, the number of buddies a student is allowed is limited to encourage tight collaboration and good buddy choices. Only the messages for one topic, consisting of a dialog, are displayed at one time. Messages are displayed in the window on the right, while the list of topics accessible to the student is displayed on the left. Students can respond to any message in the selected topic and can also create new topics to start new dialogs.

The QA application facilitates student-teacher interactions by providing a forum for asking questions to the teacher. The teacher can answer questions from any session and direct her responses to the student who answered the question specifically or broadcast the question-response pair to all students using the application. In the QA application, the teacher also

has the ability to send messages to one or more students without having to respond to a particular question.

III. I-MINDS Statistical Analysis Tools

The BG and QA applications have different analysis tools: The BG application includes the DatabaseSearch (Figure 3) analysis tool with two embedded features. The first feature allows the student to graphically view the dialog for any message. The second embedded feature uses AI techniques to keep track of user dialog involvement in each topic. This feature alerts the user when a non-buddy is contributing to a topic the user is interested in and suggests adding him as a buddy, when the user's contribution level is too low, or when the agent believes that there are too many unread messages in a specific topic.

The teacher version of the QA application consists of the VirtualClassroom analysis tool (Figures 1 and 2) with an embedded feature using machine learning and natural language processing. This feature parses the questions received from students into keywords and assigns the question a score based on weights stored for each keyword. The weights are increased or decreased based on whether the teacher chooses to answer or ignore questions containing those keywords. Questions are also sorted by score in the QA application.

Virtual Classroom

The VirtualClassroom (VC) analysis tools in I-MINDS are designed to assist the teacher in synchronously monitoring the students using the CSCL system. Figure 1 shows the VC interface. The statistics collected from user interactions are presented using a pie chart for each student. The teacher can set thresholds for each statistic individually, customizing the VC based on her expectations for the classroom environment. For example, modifying the thresholds for the responsiveness statistics assists the teacher in identifying the strengths and weaknesses of each student. The wedges for this pie chart are colored blue when the statistics are exceeding these thresholds and red when they fall below them. The messages sent by I-MINDS users displayed on the right side of the screen are filterable by the session history. By selecting any message, the teacher can view the message tree associated with it.

The statistics are also presented in a visually convenient manner using bar graphs. The teacher can select any set of the statistics for one student or all the students for a particular statistic. The bars are colored appropriately based on the threshold values, just like the pie charts. Figure 2 shows an example of the charts created by VC for the statistics. VirtualClassroom can be updated in a fraction of a second by re-sending the queries to the database. The analysis tools also intelligently process the data, looking for trends where the majority of students fail to meet the threshold for one or more statistics. If such a trend is found, the teacher is alerted with a popup window as shown in Figure 1.

DatabaseSearch

The DatabaseSearch analysis tool is designed to assist users in viewing previously sent messages. The only difference be-

tween the student and teacher versions of DatabaseSearch is the constraint that students can only view messages from their buddies while the teacher can view messages from any user. Figure 3 shows the interfaces for DatabaseSearch which provides the user with numerous metadata filters for message viewing. Messages can be viewed by content, sender, message type, session interval and timestamp received interval. Messages matching the filters are displayed in a list. The messages trees for any particular message can also be viewed as shown in Figure 3. The user tab for DatabaseSearch (not shown) displays the number of messages associated with a particular user and the buddies of that user.

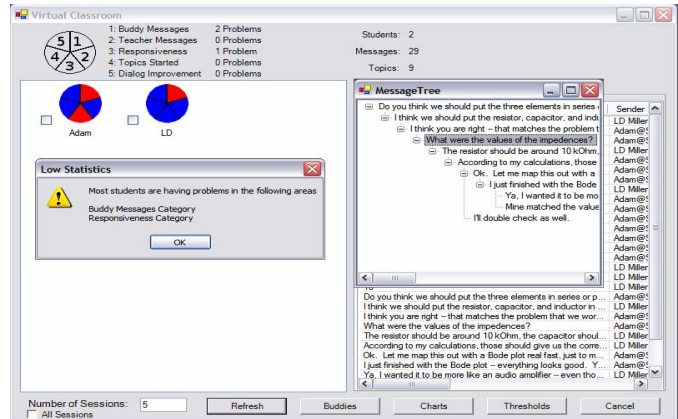


FIGURE 1  
VIRTUAL CLASSROOM ANALYSIS TOOL FOR I-MINDS

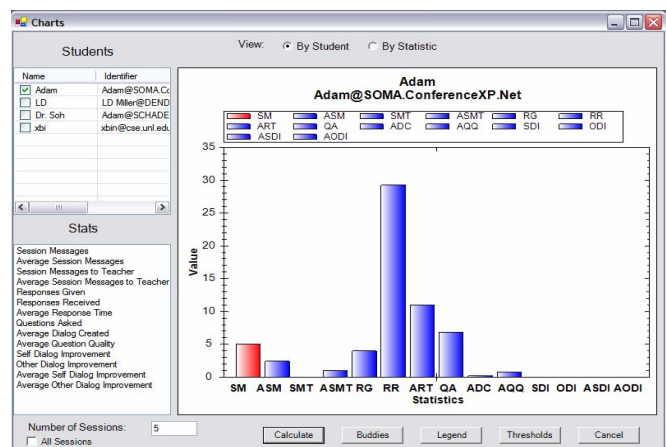


FIGURE 2  
CHART CREATED BY THE VIRTUAL CLASSROOM

DEPLOYMENT

We first deployed CXP+I-MINDS to Bellevue University in the summer of 2006, where it was used during the instruction of a business course consisting of around twenty students. CXP+I-MINDS received favorable reviews for both its interface and functionality from the students and teacher. The instructor has said that “[I-MINDS] has provided [him] with solutions for improving [his] classroom activity that were lacking in other options that [he has] explored.”

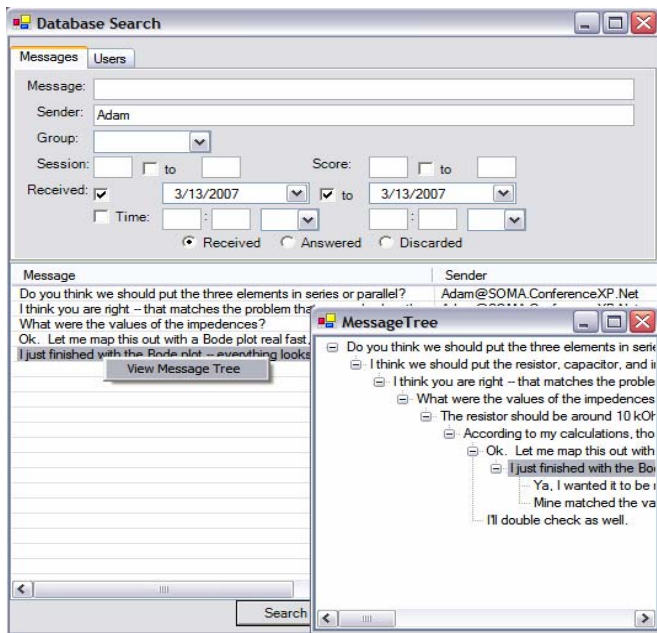


FIGURE 3  
DATABASESEARCH ANALYSIS TOOL FOR I-MINDS

**CONCLUSIONS AND FUTURE WORK**

We have proposed a design paradigm consisting of five principles for creating analysis tools in CSCL systems, based on the collaborative management paradigm [3]. These principles—which include generating content and relational information for tracked data; displaying the statistics in an organized, informative manner; allowing user customization in the analysis tools; and offering automated support through intelligent data processing in both a synchronous and asynchronous environment—guided the design of our analysis tools for CXP+I-MINDS: VirtualClassroom and DatabaseSearch.

Table I shows a comparison of CXP+I-MINDS with the CSCL systems discussed in the related work. We observe that only DEGREE and I-MINDS automatically track relational and content information about user data and possess the capability to process data intelligently. All of the tools provide a visual display of information to users to help with decision making, and all but ColAT allow for both synchronous and asynchronous operation of the analysis tools. Overall, only I-MINDS meets all the design principles.

Presently, we continue to refine the statistics provided by CXP+I-MINDS to offer more information to users and generate more advanced user models to improve the support offered by intelligent information processing. Moreover, we will work to add support for the other modes of operation (e.g., audio/video, shared whiteboard) to our analysis tools. Finally, to improve user customization, it would be useful to design a system where users can define new statistics that can automatically be tracked and computed without having to redesign the application.

TABLE I  
COMPARISON OF ANALYSIS TOOLS

	C/R Info	Display Info	User Custom	Intel Process	A/S Ops
Synergo	-	+	+	-	+
ColAT	-	+	+	-	-
DEGREE	+	+	-	+	+
I-MINDS	+	+	+	+	+

**ACKNOWLEDGMENT**

The authors would like to thank David Levy at Bellevue University and the members of the ConferenceXP development team at Microsoft Research, specifically Tim Chou and Jason Van Eaton. This project is sponsored partially by a Microsoft Research ConferenceXP grant and a Pepsi sponsored UCARE grant at the University of Nebraska-Lincoln.

**REFERENCES**

- [1] Beavers, J., Chou, T., Hinrichs, R., Moffatt, C., Pahud, M., Powers, L., and Van Eaton, J., "The learning experience project: enabling collaborative learning with ConferenceXP", *Microsoft Research Technical Report MSR-TR-2004-42*, Apr. 2004
- [2] Liu, X, Zhang, X., Soh, L.K., Al-Jaroodi, J., and Jiang, H., "I-MINDS: an application of multiagent system intelligence to on-line education", *Proc. IEEE Int. Conf. Systems, Man, & Cybernetics*, Oct 5-8, 2003, pp. 4864-4871
- [3] Jermann, P., Soller, A., and Muehlenbrock, M. "From mirroring to guiding: A review of state of the art technology for supporting collaborative learning", *Proc 1st European Conf. Computer-Supported Collaborative Learning*, Maastricht, The Netherlands, March 22-24, 2001.
- [4] Barros, B., Verdejo, M.F., "Analysing student interaction processes in order to improve collaboration. The DEGREE approach", *Int. J. Artificial Intelligence in Education*, Vol. 11, 2000, pp. 221-241.
- [5] Avouris, N., Komis, V., Fiotakis, G., Margaritis, M., and Tselios, N., "A tool to support Interaction and Collaboration Analysis of learning activities", *Proc. 6th Computer-Based Learning in Science*, July 2003, pp. 215-225.
- [6] Voyiatzaki, E., Margaritis, M., Avouris, N., "Collaborative Interaction Analysis: The Teachers' Perspective", *Proc. 6th IEEE Int. Conf. Advanced Learning Technologies*, July 2006, pp. 345-349
- [7] Dimitracopoulou, A, "Designing collaborative learning systems: Current Trends & Future Research Agenda", *Proc. Int. Conf. Computer Supported Collaborative Learning*, 2005, pp. 115-124.