

Online Collaborative Learning Enhancement through the Delphi Method

Prof. Murray Turoff
Prof. Starr Roxanne Hiltz
Xiang Yao
Information Systems Department
<http://is.njit.edu>
New Jersey Institute of Technology
Newark NJ, USA
Email: turoff@njit.edu, hiltz@njit.edu, Xy3@njit.edu

Dr Zheng Li
Department of Information Systems
Pace University
New York, USA
Email: zli@pace.edu

Dr Yuanqiong Wang
Department of Computer and Information Sciences
Towson University
Maryland, USA
Email: kwang@towson.edu

Dr. Hee-Kyung Cho
Department of Information Systems and Decision Sciences
Silberman College of Business
Fairleigh Dickinson University
Teaneck NJ, USA
Email: hkc7959@njit.edu

Abstract

A variety of field trials have been conducted at NJIT in the past few years to demonstrate the utility of a Delphi-like approach to promoting asynchronous class wide collaboration. These utilized the Social Decision Support System (SDSS) originally developed as a Computer Mediated Communication (CMC) system for large group decision support. This paper provides an overview of these studies and then focuses on a recent case study in the fall of 2003 that demonstrated the ability of a computer mediated asynchronous Delphi process as a tool to scaffold collaborative idea generation and evaluation in both face to face and distance courses.

Keywords

GDSS, GSS, CMC, SDSS, Delphi, eLearning, ALN, collaborative learning, Knowledge systems, system security, distance learning, online learning, idea generation

INTRODUCTION

The National Center for Education Statistics (2003) estimates that there are currently over 2 million students in the US alone taking online courses, and there are millions more in other countries. A growing unknown number of students are now using CMC systems to augment regular face to face courses by extended discussion outside of the classroom (Hiltz and Goldman, 2004). Discourse, a form of collaborative learning (Vygotsky 1980), is central to learning in this environment. The threaded discussion is the prevailing mode of organizing collaborative discourse, but it tends to become disorganized and confusing when large groups of students work on a complex issue over a period of weeks. More robust methods of scaffolding such collaborative discourse are necessary. This paper describes the use of a variation of the Delphi method to scaffold complex discussions in Asynchronous Learning Networks (ALNs, see <http://www.alnresearch.org>). The objectives of the cases reported here are idea generation and evaluation. The most prominent source for recent work in this area is idea generation in Group Decision Support Systems (GDSS).

In previous GDSS research (Dennis, 1993, Gallupe, et. al 1992), the effectiveness of group idea generation was evaluated mostly in a decision room environment in which group members generated ideas with computer technology support. These idea generation processes were synchronous in terms of time and place and the contributions of group members were made by a combination of verbal and text modes. However, this stream of research did not make clear whether the effectiveness of idea generation was due to the effect of technology itself

or the modified structure given by technology (Pinsonneault, Barki, Gallupe, and Hoppen 1999). In recent studies, the focus of research was shifted from these traditional comparative studies between GDSS and face-to-face groups into the evaluation of specific idea generation techniques (Santanen, Briggs, and Vreede, 2004, Garfield, Taylor, Dennis, and Satzinger 2001). Our own work at NJIT has focused on asynchronous oriented GDSS and recently on computerized forms of the Delphi Method (Linstone and Turoff, 1975). This orientation is towards larger size groups (e.g. class size) and knowledgeable participants – traditionally experts, but in this case students who have been immersed in learning the topic.

The Delphi Method was created in the 1950's at the RAND Corporation to allow large groups of experts to contribute collectively to the examination of complex problems (Linstone and Turoff 1975). In fact, the first Delphi application was the replacement of a computer simulation by a process of subjective estimations by large groups of experts (Dalkey and Helmer, 1951). Since it was often used for predicting the future occurrence of technological breakthroughs it acquired the name of Delphi (after the Greek oracle at Delphi). However, the general concept of the technique (Linstone and Turoff, 1975) is:

Delphi is a process of structuring a written, asynchronous communication process among a large problem solving group so that it is tailored to the nature of the problem, the characteristics of the group, and the objectives of the problem solving exercise.

The Delphi process uses phases of a collaborative problem solving process such as exploration, understanding, and evaluation to structure the communication process. Computer based versions allow participants to participate in any phase at any time and eliminates sequential constraints for each individual. It allows large groups of students to engage in the process and for each individual to focus on what they wish to deal with at any time of their choosing. The results of this can:

- Improve idea generation
- Self organize the contributed content
- Facilitate equal participation of all students
- Reduce information overload problems for large classes
- Facilitate collaborative problem solving
- Utilizing voting to focus discussion on areas of disagreement and uncertainty
- Facilitate understanding by enhanced visualization through the use of scaling methods
- Expose disagreements for focusing the discussion
- Facilitate comprehensive idea evaluation
- Allow exchange of tacit knowledge among professionals.

BACKGROUND

In 2003 two Group Decision Support Systems (GDSS) Ph.D. thesis efforts (Li 2003; Wang 2003) involved creating and evaluating through experimentation with a Delphi-like (Linstone and Turoff 1975; Turoff and Hiltz 1995) software system we refer to as a Social Decision Support System (SDSS) (Turoff, Hiltz, Cho, Li and Wang 2002). This CMC system was designed to allow large groups of people (e.g. hundreds) to address complex issues such as the relative value of any group of related items (e.g. tasks, goals, budget allocations, criteria, etc.).

Common to almost any type of problem situation is the initial step of compiling one or more lists of contributions of a given type. For example, in a typical decision problem the types of lists that are usually needed are (Turoff 1990):

Type of List	Voting Scales	Examples
Objectives	Importance, Measurability	Getting a college education, Goals of an organization
Criteria	Importance, Measurability	For a presidential candidate, Buying a house, Choosing a college
Solutions, Actions, Decisions, Policy Resolutions	Desirability Feasibility	Components of a national medical drug program Potential products for a company
Consequences, Impacts, Inferences, Events	Likelihood, Significance	Long term impacts of modern terrorism on U.S. Society
Observations, Viewpoints, Pro/Con Arguments, Rationales	Validity, Importance	Reasons for the U.S. Civil War, Scenarios of any type, Rationales
Interpretations, role playing	Validity, Significance	How would a selection of ex-presidents reflect on a current political issue?

Table 1: Examples of typical list types

The possible types of list generation that one can treat as a Delphi process are completely open ended. Once these lists have been created the natural extension is to consider interactions between lists such as the

relationship of the decision choices that can influence the entries in a list of consequences. Typically this is done through a matrix or multidimensional type interaction structure (e.g. payoff matrices). However, for the scope of this short introduction we will only focus on the creation of a list, the various possible types of lists, and the typical voting scales used to assess each list. This is summarized in table 1. Note that the usual form of the scale is an ordered categorical scale but may in some applications be rank order, interval, or ratio scales. The most general formulation of this problem is found under the subject of “multi-criteria decision problems.” These are problems involving pragmatic knowledge and ultimately subjective tradeoffs or compromises for reaching decisions.

EVALUATING COURSE OBJECTIVES

For example, the system has been used in a field study mode in five different graduate courses to allow students to propose the important things they have learned in a course and to rank order them (Wang, Li, Turoff, and Hiltz 2003). The resulting rank order rated by each student can be combined in one scale using Thurstone’s Law of Comparative Judgment to translate rank orders to a single group interval scale for the group as a whole (Li, Cheng, Wang, Hiltz, and Turoff, 2001; Thurstone 1927).

The example voting results in Figure 1 show the top ten items (out of 28 proposed) rated in one of the courses (CIS 679, Management of Information Systems). Each student was asked to suggest only the most important item he or she learned in this course provided it was different than the other items already entered by the other students. Note that the top rated item is more than two times the scale strength of the second item (16.5:6.78) and the next three items are essentially at the same scaling point, showing an equal rating for the group as a whole for all three items. This demonstrates the power of a good scaling method to provide significant visualization results that aid the group to interpret and understand what their votes mean. The usual determination of the simple group ranks is shown in the first column. This rank loses all the additional insight (degrees of consensus and disagreement) provided by the Thurstone scaling of the results. In all the case studies of this type the results were surprising to the faculty, as the results often did not correspond to their expectations. Understanding the results often required reviewing the comments made in the discussion of the items by the students. Of course the user has to be taught the meaning of an interval scale and some might claim that makes it less easy to use. Various clustering methods and multidimensional scaling also provide higher dimensional visualizations (Carroll and Wish, 1975).

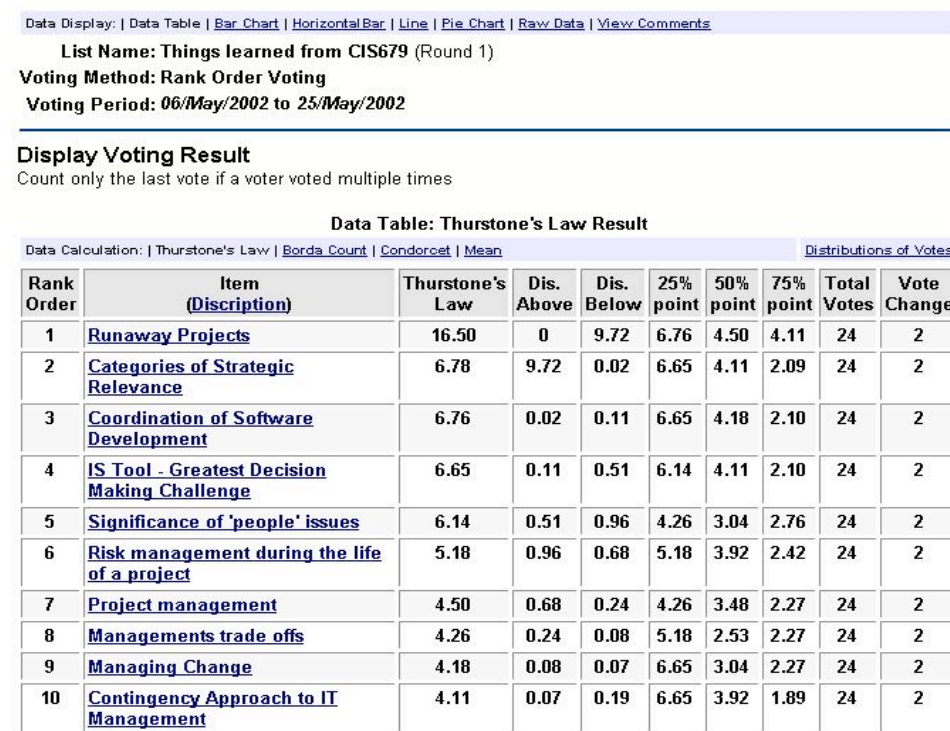


Figure 1: Voting Result for top ten items out of 28 for CIS 679

The faculty involved in the cases of this type felt that this method of evaluating the outcome of a course is far more informative for improving a course than the standard university “student satisfaction” type survey that is now commonly used. In the above case (Figure 1) the faculty member finally realized from the discussion that, while the topic of runaways had only an hour’s lecture in the whole course, many of the students were using it as a framework to organize the other material in the course. This led to moving the topic up to an earlier part of

the course, pointing out some of the relationships to later topics, and referring back to it when discussing other topics in the course. Participation in these exercises was totally voluntary, 28 out of 38 students participated in suggesting items, and 24 completed the voting process of ranking all 29 items. The exercise was over a two week period at the very end of the semester, including the last week of the course and the week of finals.

PRODUCT IDEA GENERATION

In a recent thesis experiment (Cho 2003, 2004), a plain CMC system (i.e. WebBoard) was used in a two by two factorial experiment: with or without a Delphi structure in typical CMC system, and with small (5-7 people) or medium sized groups (11-14 people). The Delphi feedback feature was added by the use of Survey Tracker™ to provide rapid feedback on voting positions at the start, during the process and at the end. Each student group was asked to act as a consulting group to recommend product ideas for a new pill-sized device that can store and emit data when triggered by a signal. This was an open ended brainstorming task with no real effective limit on the number of ideas. They had approximately two weeks to work on the task through the asynchronous CMC system. The results are very significant with respect to the positive impact of the Delphi structure as can be seen in the following table.

Structure X Group Size	Small	Medium	Total
Delphi: Raw	157	247	404
Unique	67	111	132
Rare	40	56.5	96.5
Unstructured: Raw	108	192	300
Unique	52	86	110
Rare	18.5	48	66.5
Total Raw	265	439	704
Unique	94	145	188
Rare	58.5	104.5	163

Table 2: Distribution of total Raw, Unique, and Rare Ideas (Rare ideas are defined as occurring in no more than 3 of the 44 groups)

The statistically significant results (Cho 2004) of this experiment were that the Delphi structure is more effective (statistically significant at .05 or less) in producing more total raw, unique and rare ideas. The findings of the experiment indicate that the Delphi structure helps asynchronous CMC groups to generate more unique ideas per person. The finding on the dimension of group size indicates that having more people in an asynchronous CMC group does produce more total unique ideas but not on a per person basis. However, small groups produce more ideas per person in the Delphi condition than in the unstructured condition. For this particular open ended problem the medium sized group did make statistically significant more contributions and did not run out of ideas.

The results indicate that brainstorming groups can be quite large and there are Delphi studies that have involved hundreds of respondents. Typically the rule of thumb used is to determine how many different areas of knowledge can contribute to the subject of the inquiry and multiply that number by both 3 and 5 to determine the lower and upper limit for a Delphi group. For objectives like developing a new product concept in a company, this easily can involve 50 to 100 individuals.

EVALUATION OF PREVENTIVE SECURITY MEASURES

In the evaluation of online learning in distance or in blended face to face plus online classes, collaborative learning is the pedagogical methodology that makes online learning as effective or even more effective than the standard face to face class (Hiltz and Turoff 2002; Turoff and Hiltz 1995; <http://www.aln.research.org>). This case study, completed in December 2003, is the latest in a series of three-round Delphi exercises designed to scaffold online learning experiences.

We conducted a three week exercise involving two classes and a group of 20 students who were willing to participate in the last three weeks of the semester. They used the same SDSS system used for the evaluation of course outcomes discussed previously.

CIS 681 Information Systems Security Auditing (blended face to face section with CMC support)
 CIS 679 Management of Information Systems (distance online section)

The task was to examine, discuss, and rate different preventive measures for Information Systems Security. Any participant in the exercise could at any time:

1. Propose new items for the list of items

2. Propose alternative wording of an existing item and vote yes or no to accept an alternative wording
3. Comment on any single item with replies that are classified as Pro, Con, or Neutral
4. Vote on the relative value of all the items using the chosen voting method
5. Continue to discuss with pro, con, and neutral items using the current voting results to focus the discussion
6. Change one's vote at any time as motivated by the discussion.

A dozen different types of voting processes may be employed in the current software for any list of items. We are now adding numeric input for estimating variables like budgets, degree of risk, and probabilities of success. For the “security” application we utilized the following ordered nominal voting scale:

- CI** Critically Important (e.g. an organization must do this)
- VI** Very Important (e.g. must be done given the right conditions)
- I** Important (e.g. useful if one has the funds to expend)
- SI** Slightly Important (e.g. rare situations when useful)
- UI** Unimportant (e.g. not useful at all and may be counterproductive)

Since it has been estimated that over 50% of security problems are brought about by an organization's employees we included both human and technical preventive measures. Most textbooks explain security measures but give very little guidance on how critical each one is. More measures can be employed than most organizations can afford and this was the dimension the students were asked to deal with by the above voting scale.

Results

An initial 60 preventive measures were supplied from standard text books that never really deal with the problem of how much a specific organization can afford to do. The students contributed 13 new ones and proposed 25 modifications to the existing ones. Approximately half were Masters and half were Ph.D. students. About half were working students and half were full time. Two students reported some prior work experience in security. The activity that occurred in a three week period is summarized below in Table 3.

Preventive Measures	72	Total Words (approx.)	25,000
Comments on	433	Average comment size (words)	480
Pro comments	200	Contributors	20
Con comments	104	Voters	27
Neutral Comments	129	Contributions/person	23
Modifications	25	Contributions/day	25
Total of all items	530	Comments/Measure	6

Table 3: Distribution of Contributions

It should be clear that maintaining involvement in the equivalent of 72 separate discussion threads that are continuously active for three weeks would normally be an information overload situation. If a participant was on every day this would be 25 new items a day to consider and this does not factor in the task of voting on those items. This particular case study was one with an intense level of activity (Table 3).

The end result is all 72 items with the final votes in the order of the weighted average of the voting scale. This gives only the title of the item and definitions appear in the text of the items. In Figure 2 we show the seven items rated the most important (at the top of the list) and the seven items rate at the bottom of the list to be least important. The system provides the full list to everyone.

Display Voting Result

Count only the last vote if a voter voted multiple times

Data Table: Total number of Votes

Data Calculation: Total Number of Votes Percentage										
Rank Order	Item (Description)	CI	VI	I	SI	UI	NJ	WA	Total Votes	Vote Change
1	Have an operative backup system for all servers	11	3	3	0	0	0	4.47	17	10
2	All employees should be informed of security policies.	9	7	1	0	0	0	4.47	17	10
3	Backup solution	10	4	3	0	0	0	4.41	17	10
4	Always employ an antivirus detection system	10	4	3	0	0	0	4.41	17	10
5	Formulate a formal Security Policy	7	9	1	0	0	0	4.35	17	10
6	Always use authentication and encryption for transition of sensitive data.	9	3	4	1	0	0	4.18	17	4
7	Disable all accounts and access of employees no longer serving in the company.	6	9	1	1	0	0	4.18	17	10

66	Disguise the Network with Honeypots	0	2	6	5	4	0	2.35	17	10
67	Write-protection for diskettes	0	2	4	8	3	0	2.29	17	4
68	Remember at least 10 passwords of the user	0	3	4	5	5	0	2.29	17	10
69	Encrypt all data sent over then internal and external networks.	2	1	6	1	5	2	2.29	17	10
70	biometrics authentication	0	3	1	10	3	0	2.24	17	4
71	Install keystroke loggers on all console keyboards	0	2	2	6	6	1	1.88	17	10
72	Install keystroke loggers on all workstations	0	0	2	6	9	0	1.59	17	10

Figure 2: Voting results for first and last seven items

The voting results show that there was a high degree of differentiation among the preventive measure items by the students. At least half of the vote changes recorded are by those who took an initial “no judgment” position on many of the items. No one could see the vote result until they had voted. They were advised not to vote on an item, if they had no confidence in their choice. This is a typical Delphi instruction even when done with expert groups.

Most educators assume that collaborative learning means you break the class into small teams. A different hypothesis that many of us who utilize class-wide collaboration as a learning tool have is that the discussions are a significant learning experience. In this case we observed that knowledge was being conveyed from those students with relevant experience to those without it, and that the level of vote changes was significant.

A very significant factor in the design of this system is the use of voting to focus discussion on the items that exhibit disagreement or uncertainty. We had students in many of the trials of the software initially express the view that they thought voting was a one time thing with respect to any specific vote, which shows a learning curve for the use of dynamic voting.

The use of a continuous voting process (dynamic voting) is a dramatic difference made possible by moving the Delphi method from the paper and pencil environment to the online environment. What this also allows is the ability of the members of the group to choose what phase of the Delphi process and what specific content items they want to deal with at any time in the whole process. This is viewed as a “reciprocal” group process since a change by one member with respect to adding comments or changing votes may cause or influence others to reciprocate (Hiltz and Turoff, 1993). There is no longer any need to impose a sequential group process on the procedure. Every member of the Delphi process is free to engage in individual cognitive problem solving process with little no constraints imposed by the group process.

Discussion Examples

The concepts of using voting initially to focus discussion, and scaling methods to promote understanding, are key elements of the Delphi process that leads to collaborative efforts that would otherwise degenerate due to information overload in CMC systems (Hiltz and Turoff, 1985). Given the varied mix of students with respect to actual work experience and a few with some experience in security concerns in their companies, we observe a significant flow of information from those with such pragmatic knowledge to those without it, and this in turn influenced the occurrence of vote changes. The termination of a Delphi exercise of this type results from the frequency of vote changes decreasing to essentially zero. The following is an excerpt from the discussion about just one of the 72 items. Student names have been removed.

Root Item: Audit Trails

By: Student One (03:05PM, 11/24/2003)

Audit trails should be implemented within the organization. The audit trails would allow review of activities leading up to a particular event. Standard logging such as provided by web server logs should also be augmented with application logs. All applications should be required to implement audit trails and have that functionality turned on. This would allow for accountability. Note that this is not a preventive security measure. Security measures are classified as preventive, detective, or corrective.

Neutral: Audit trails

By: Student Two (12:57PM, 11/25/2003)

I am neutral on this, because as Student One said this is not a preventive measure, but falls into detective category. Logs could be expensive in terms of storage and performance. So the state of the budget would probably influence this measure to be employed in a company.

Pro: This can be a good idea with some constraints...

By: Student Three (07:56PM, 12/03/2003)

This type of logging can prove to be very useful in the event of an attack. However, left unchecked it can also be a huge burden to the system. A good method to make this practical would be to dump all the logs off the host system onto a tape backup at the end of each week. A three-week rotation can be used before the first week’s logs get overwritten. This would provide a three week trail of logs leading up to any attack, and

would also minimize the overhead involved with data storage, and at most keep a weeks worth of logs on the host system (to prevent bogging it down)

Pro: If the resource and space is available then it is

By: Student Four (08:40AM, 12/08/2003)

This can help in problem determination. There should be security measures in place that should prevent any sort of attack from happening in the first place. But since human ingenuity will find a way around anything, there is no such thing as total 100% security. So in that case this would be a nice thing to have, but in no way should this be considered more important than the security measures that can prevent attacks. If resources of people, space and time were available then implementation of this would be very useful if there is an attack.

Pro: Given Student three's Suggestion I would work

By: Student Two (11:09PM, 12/03/2003)

Student three's suggestion definitely gives a workable solution for this measure. I guess I just turned pro for this one. But would still probably keep it as 'important' measure, not the VI or CI.

Neutral: I agree the importance, but have to be neutral....

By: Student Five (12:10AM, 12/04/2003)

There are many reasons to use full audit trails. Like the federal code on clinical trials that all clinical data points must be audit trailed with date time stamp and user identity. But implementing in any fields of data in the system is no way to accomplish that requirement. There will be a secured fortress but a performance bear.

The discussion transcript makes it very clear why items moved to the top or bottom of the list. One surprise was the low rating of the preventive measure on "biometric authentication," but the following discussion entry was very typical of what a lot of students said and why the resulting item went to the bottom of the list.

Neutral: Biometrics is not a panacea for all problems

By: XXXXXXXXXXXX (11/25/2003)

While Biometrics has a lot of promise, it is still not yet here. Most of the biometric systems installed during the biometric hype days after 9/11 are begin de-installed. People thought biometric systems would do everything; hence they installed face recognition systems on airports to spot terrorists, systems on the sport grounds to spot hooligans, etc. Two years later not a single culprit has been nailed by this system. So while I am a big fan of Biometrics, there is need for caution. Each of the authentication methods like token, knowledge and biometrics systems have areas where they are strong and other areas where they are weak. It is up to you as the manger to decide after consultation on the appropriate choice and remember you have to justify your actions.

It is also clear when one looks at the discussion, that a number of students chose a neutral classification for comments which were perhaps more on the "con" side and this might have been due to the lack of anonymity on the discussion and also not to be confrontational. The actual software allows anonymity to be set for all participants but we did not use that feature in this exercise.

This case allowed a group of 20 students to carry out the equivalent of a three round Delphi process in three weeks to generate and evaluate preventive measures to ensure system security. This involved creating and modifying a list of items as well as being able to vote and revote dynamically on the relative importance of the items while carrying on a continuous discussion of the merits of the items. Attempting to do this with any of the standard commercial software for online education would easily result in information overload before such a result could be obtained.

INTERACTION DESIGN

Select a Topic: Refresh Topic
CIS679 Discussion 2003F (New)

Contribute: All
Root Item
Comment
Modification

View: All
Root Items
Comments
Modifications

Mark Read: All
Root Items
Comments
Modifications

Vote: Root Items
Result
Modifications

Other: Search
Help
Administrator

Logout

Root Items

Type	#	New	Votes Needed
Root Items	72	72	Vote Result
Mods	25	25	25
Comments	419	419	N/A

Detailed information about the Root Items, Modifications and Comments:

Item: [Edit](#) | [Delete](#) | [Move](#) | [Mark Read](#) | View: [Comments](#) | [Modifications](#) | [Contribute](#): [Comment](#)

Item: Audit Trails

By: Ron Singh (03:05:04PM, 11/24/2003)

Audit trails should be implemented within the organization. The audit trails would allow review of activities leading up to a particular event. Standard logging such as provided by web server logs should also be augmented with application logs. All applications should be required to implement audit trails and have that functionality turned on. This would allow for accountability. Note that this is not a preventive security measure. Security measures are classified as preventive, detective, or corrective.

Comment: [Edit](#) | [Delete](#) | [Mark Read](#)

Pro: If the resource and space is available then it is

By: Barun M. (08:40:19AM, 12/08/2003)

This can help in problem determination. There should be security measures in place that should prevent any sort of attack from happening in the first place. But since human ingenuity will find a way around a thing, there is no such thing as total 100% security. So in that case this would be a nice thing to have in no way should this be considered more important than the security measures that can prevent attack. If resources of people, space and time is available then implementation of this would be very useful if there is an attack.

Comment: [Edit](#) | [Delete](#) | [Mark Read](#)

Neutral: I agree the importance, but has to be neutral...

Figure 3: Main interaction (Strategic) screen

The initial screen (Figure 3) provides at the top all the strategic commands (navigation menu) that immediately take the user to an action screen such as creating items, viewing or inputting votes, etc. This is a completely broad menu and it provides complete comprehension early on. The lower left column scroll provides immediate information on what is new and a status table of the different types of items. By clicking on any cell in the status table the user calls up a list of those items into this column (current list summary). The column on the right is the primary working page. By clicking on a root item on the left, one can get the complete discussion thread; or by “viewing all” choice from the navigation menu all items may be called up and displayed in detail. The navigation menu and the “Root Items” status table are always visible to the user.

Using the context to provide reactive commands beyond the navigation menu above is what makes the system very easy to learn. The user may choose to modify, edit, mark read, or propose alternatives to the item they are looking at in the linear menu that appears above the given item. The different types of items were distinguished by highlighting color, indentation structure, and labels. Anywhere in the interface, clicking on the counts of items brings up a display of the full text of the items. When the user chooses the strategic action to vote (or revoke), the following screen (Figure 4) is provided in the working area in the right scroll. The user is shown current votes and allowed to change them at anytime.

My Voting History | **Voting Result**

List Title: IS Security Measures (Round 4)

Vote Method: Categorical Voting

Please vote on the following 72 list items:

Note: The data displaying is your last voting result.

Deadline: Aug-30-2004 12:00:00AM [Reset](#) [Submit Vote](#)

No.	Item	Vote Here
1	Fire protection for computer rooms	No Judgment
2	Not locating computer rooms in a basement or first floor	No Judgment
3	Using Integrity Checkers	Critically Important
4	Separation of responsibility (developer vs. maintenance)	Very Important
5	Backup solution	Important
6	Protect network devices updates	Slightly Important
7	Encryption of executable files	Unimportant
8	Create a special master boot records.	No Judgment
9	Use anti-virus software on workstations and servers	No Judgment

Figure 4: Voting Screen

The system provides the ability to bring up a complete transcript of the total discussion, which can be scrolled or printed. We feel this is an important feature for those who want to make a printout and/or comparatively examine material that is associated with different root items. Email approaches which force users to look at only one item at a time, or some CMC systems that restrict the user to only viewing one discussion thread, in effect

inhibit the evolution of complex discussions and group activity by breaking up short term memory with too many interaction operations.

In all our studies the typical student was using WebBoard® or WebCT® for courses at the same time that the Delphi Discussion software was used. We have observed no lowering of satisfaction measures for the system compared to use of these other CMC systems (Li 2003; Wang 2003). Since it is more typical to get a lowering of satisfaction for the first use of a new system, we interpret this as a positive factor in the design of the interface. With respect to the preventive measures Delphi it was quite clear that the students had a serious interest in the subject and that it was one of real current concern in their organizations.

SUMMARY

Our contributions are a methodology and a software structure to scaffold complex CMC discussions by large groups of students learning together online. The field trials and experiment reported demonstrate that Delphi-like structures can support large groups engaged in complex collaborative problem solving discussions.

More work is needed on the content analysis of these types of discussions to obtain quantitative measures of the amount and effectiveness of knowledge transfer and learning. Content analysis relating discussion entries to changes in voting would give us further insight into the hypothesized relationship. Another area of investigation is to attempt much larger groups than the 20-40 range we currently have been working with. We are also planning use of the SDSS system with professional groups on similar topics related to systems security, since it is obviously also a collaborative knowledge gathering tool. This will certainly involve larger groups.

This particular software design was implemented as a specific structure to allow very large groups to agree on a list or a number of lists of specific items and rate their relative value. There are literally hundreds of possible designs (Linstone and Turoff, 1975; Turoff and Hiltz 1995). The next step with this structure is to allow semantic links between items in different lists. For example, a decision alternative in one list may be linked to the specific consequences it has in a separate list of consequences. This would be a first step in moving to a collaborative structural modelling system.

The design structures developed in this approach create the ability for large groups, with facilitation guidance, to exchange knowledge on a continuous basis as a regular ongoing activity. In the case of complex large group collaboration it is critically important to minimize cognitive overhead and to provide the ability for participants to handle very large transcripts of material. The use in this case of (1) comprehensive, broad, one level interaction menu, and (2) context visibility to provide the submenu choices, are clearly two of the approaches that make an impact on reducing cognitive overhead so the participants may concentrate on the discussion and not the mechanics of the interface.

As online applications become more encompassing of all human intellectual activities, the natural evolution of HCI has been one of sweeping application domains into the interaction design challenge. Nowhere is there a greater challenge than when dealing with alternative CMC group communication structures.

REFERENCES

- Carroll, Douglas, Wish, Myron (1975) Multidimensional Scaling: Models, Methods, and Relationships to Delphi, in *The Delphi Method book*: see Linstone and Turoff, 1975 below.
- Cho, H. Delphi Structure and Group Size in Asynchronous Computer Mediated Communications. *Proceedings AMCIS 2003*.
- Cho, H. (2004) *The effect of Delphi structure on small and medium-sized asynchronous groups*, Ph. D. Dissertation, January 2004, New Jersey Institute of Technology, Information Systems Department. <http://www.library.njit.edu/etd/index.cfm>
- Dalkey, N., Helmer, O., (1951) The Use of Experts for the Estimation of Bombing Requirements – A project Delphi Experiment, *Rand Report RM-7272-PR*, November 1951.
- Dennis, A. R. (1993) Computer Brainstorms: More Heads are Better than One, *Journal of Applied Psychology* (78:4), 531-537.
- Gallupe, R. B., Dennis, A. R., Cooper, W. H., Valacich, J. S., Bastianutti, L. M. and Nunamaker Jr., J. F. (1992) Electronic Brainstorming and Group Size, *Academy of Management Journal* (35), 350-369.
- Garfield, Monica J., Taylor, Nolan J., Dennis, Alan R., Satzinger, John W. (2001) Research report: Modifying Paradigms - individual differences, creativity techniques, and exposure to ideas in group idea generation, *Information Systems Research*, (12:3), pp. 322-333.
- Hiltz, S. R., and Goldman, R., eds., (2004) *Learning Together Online: Research on Asynchronous Learning Networks*, Mahwah, NJ, Erlbaum.
- Hiltz, S. R., and Turoff, M., (2002) What Makes Learning Networks Effective?, *CACM*, (45:4), April, 56-59.
- Hiltz, S. R., and Turoff, M., (1993) *The Network Nation: Human Communication via Computer*, MIT Press, revised edition (original 1978 Addison Wesley).

- Hiltz, S. R. and Turoff, Murray, (1985) Structuring Computer-Mediated Communication Systems to Avoid Information Overload. *Communications of the ACM*, (28:7), July, 680-689.
- Li, Z., Cheng, K., Wang, Y., Hiltz, S. R. & Turoff, M. (2001) Thurston's Law of Comparative Judgment for Group Support, in *Proceedings of AMCIS 2001*
- Li, Z., (2003) *Design and Evaluation of a Voting Tool in a Collaborative Environment*, Ph.D. Dissertation, New Jersey Institute of Technology, Information Systems Department.
<http://www.library.njit.edu/etd/index.cfm>
- Linstone, H. and Turoff, M. (editors) (1975) *The Delphi Method: Techniques and Applications*, Addison Wesley Advanced Book Program. (Online version can be accessed via <http://is.njit.edu/turoff> last accessed 8/22/04)
- Pinsonneault, A., Barki, H., Gallupe, R. B. and Hoppen, N. (1999) Electronic Brainstorming: The Illusion of Productivity, *Information Systems Research* (10:2), 110-133.
- Santanen, Eric L., Briggs, Robert L., Vreede, Gert Jan De, (2004) Causal Relationships in Creative Problem Solving: Comparing Facilitation Interventions for Ideation, *Journal of Management Information Systems*. Armonk: (20:4), 167
- Thurstone, L. L., (1927) A law of comparative judgment, *Psychological Review*, (34), 273-286.
- Turoff, Murray, (1990) Computer Mediated Communication Requirements for Group Support, *Journal of Organizational Computing*, (1:1).
- Turoff, M. and Hiltz, S. R., (1995) Computer Based Delphi Processes. In: Adler, M. and Ziglio, E. (Eds), *Gazing Into the Oracle: The Delphi Method and Its Application to Social Policy and Public Health*, London, Kingsley Publishers, 55-88.
- Turoff, M., and Hiltz S. R. (1995) Software Design and the Future of the Virtual Classroom. *Journal of Information Technology for Teacher Education*, (4:2), 197-215.
- Turoff, M., Hiltz, S. R., Cho, H., Li, Z. and Wang, Y. (2002) Social Decision Support Systems (SDSS). Proceedings of the 35th Annual Hawaii International Conference on System Sciences (HICSS 2002).
- Vygotsky, L.S. (1980) *Mind in Society: The development of higher psychological processes*. Cambridge: Harvard University Press, 1980.
- Wang, Y., Li, Z., Turoff, M. and Hiltz, S. R. (2003) Using a Social Decision Support System Toolkit to Evaluate Achieved Course Objectives. *Proceedings AMCIS 2003*.
- Wang, Y., (2003) *Design and Evaluation a List Gathering Tool in a Web-based Collaborative Environment*, Ph.D. Dissertation, 2003, New Jersey Institute, Information Systems Department.
<http://www.library.njit.edu/etd/index.cfm>

ACKNOWLEDGEMENT

This research was partially supported by the New Jersey Center for Pervasive Information Technology through a grant from the New Jersey Commission on Science and Technology, and the National Science Foundation (under CISE –ITO 9732354 and grant #9818309), and the Alfred P. Sloan Foundation. The opinions expressed in this paper are solely those of the authors and do not necessarily reflect those of the sponsors.

COPYRIGHT

Murray Turoff, Starr Roxanne Hiltz, Zheng Li, Yuanqiong Wang, Hee-Kyung Cho, Xiang Yao © 2004. The authors assign to OZCHI and educational and non-profit institutions a non-exclusive licence to use this document for personal use and in courses of instruction provided that the article is used in full and this copyright statement is reproduced. The authors also grant a non-exclusive licence to OZCHI to publish this document in full in the Conference Papers and Proceedings. Those documents may be published on the World Wide Web, CD-ROM, in printed form, and on mirror sites on the World Wide Web. Any other usage is prohibited without the express permission of the authors.