

Information Visualization for Collaborative Computing

A prototype tool classifies output from an electronic meeting system into a manageable list of concepts, topics, or issues that a group can further evaluate. In an experiment with output from the GroupSystems electronic meeting system, the tool's recall ability was comparable to that of a human facilitator, but took roughly a sixth of the time.

Hsinchun
Chen

Jay
Nunamaker
Jr.
University of
Arizona

Richard
Orwig
Washington
State
University

Olga Titkova
Intel Corp.

Information technology continues to generate increasing amounts of data, most of which is useless without scalable methods to collect, analyze, process, and understand it.¹ Visualization is a promising approach to such systemization because it lets users see underlying processes and guide process simulations interactively. However, visualization must be combined with some way to make repositories of text documents more manageable, providing users with a flexible, interactive environment in which to access them.

Groupware tools have attempted to provide this type of support, but they continue to suffer from major problems. The first is information overload. Users of electronic meeting systems can generate hundreds of text lines in less than an hour, and often find it overwhelming to later browse that many comments. A popular groupware database could generate thousands of notes within a few weeks.

Another problem is vocabulary differences. Users may disagree on a term's definition or decide that two terms have the same meaning.

Finally, although tools typically run on Windows or X Windows, the screen continues to be largely text-based. Trying to scan hundreds of lines 30 lines at a time is not conducive to rapidly synthesizing key thoughts.

In this article, we describe a prototype tool that addresses these problems for GroupSystems, an electronic meeting system developed at the University of Arizona and installed at more than 1,500 business, government, and university settings. The tool automatically categorizes information, statistically clusters similar documents, and displays the organized document set graphically, providing more at-a-glance information than a typical text-based display. Users can thus more easily browse document collections.

The tool has two main aspects:

- *Text analysis (categorization).* Text analysis techniques aim to identify descriptors and develop an unambiguous internal representation of a docu-

ment. Our tool maps a collection of comments or notes into a two-dimensional grid, placing similar documents close to each other. It also determines the topic or category that characterizes each cluster of similar documents.

- *Visualization.* The tool displays the graphical image of the map and uses it as an interface to provide users with different functionality. Its use of general-purpose color and layers makes the map more accessible to browsers.

We have used the tool to analyze the output of a collaborative session with GroupSystems. The sidebar "Groupware and Collaborative Computing" describes the characteristics of electronic meeting systems like GroupSystems in more detail. We are also exploring a similar approach to categorizing and analyzing Internet home pages² and archival and corrective databases, such as nuclear plant or truck repair records.

TEXT ANALYSIS

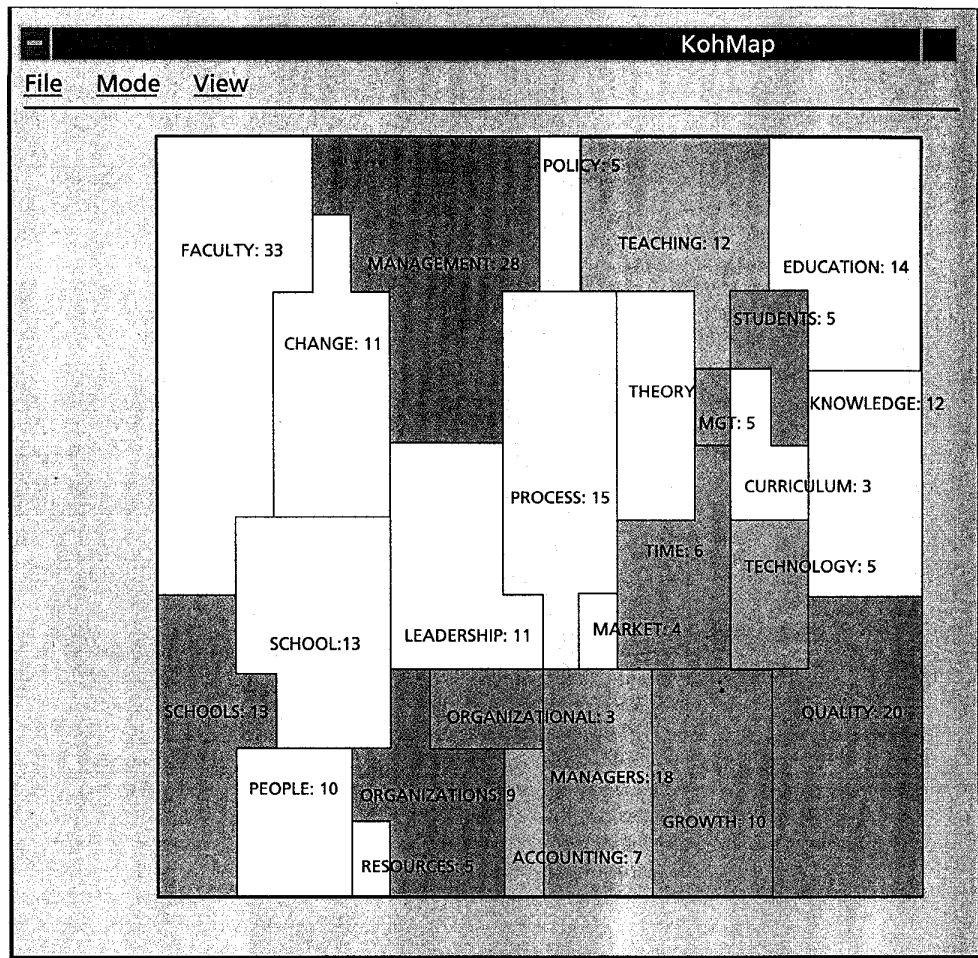
Central to our tool's approach is the *self-organizing map*, a text analysis algorithm developed by Tuevo Kohonen.³ The algorithm produces a 2D or 3D data map³ that contains two layers of nodes—an input layer and a mapping (output) layer. Each node in the input layer represents one input document (text item). After the tool processes all the input (usually after hundreds or thousands of repeated presentations), the algorithm produces a spatial representation of the input data organized into clusters of similar (neighboring) regions.

Several researchers have used the self-organizing map for text analysis and classification.^{4,6} Our approach most closely approximates the multilayered system developed by the University of Arizona's Artificial Intelligence Group to categorize Internet home pages.²

Our version of the self-organizing map algorithm consists of three steps:^{2,3}

- *Initialize the input and output (mapping) layers.* Each input and map node is a vector whose length

Figure 1. A map of data from a GroupSystems' electronic brainstorming session. In one hour, participants generated several hundred comments, attempting to answer "What issues are important to society in general and the management of organizations in particular, and what are their implications for management, education, and research?" The tool uses the self-organizing map algorithm to generate a graphical image of the results as regions and neighborhoods. The topic name is followed by the number of comments and its region sized accordingly. The use of a color spectrum (dark blue \Rightarrow rose) alerts the user to cold topics (discussed early in the meeting) and hot issues (most recently discussed).



equals the number of indexed terms in the documents. The vector for each input node is a set of values that represent the existence (1) and nonexistence (0) of indexed terms. The vector for each map node represents a weighted value of that node's indexed terms relative to node values for all the input documents. The algorithm initializes each map node's weight to a small random value.

- *Present input.* The algorithm presents an input vector and computes the distance between this vector and each mapping node vector. The node with the minimum distance measure is the "winning node" and is selected as the center of a neighborhood of nodes surrounding it.
- *Adjust weights in a neighborhood.* The algorithm adjusts the weights of the winning node as well as

Groupware and Collaborative Computing

Groupware includes a range of applications, such as e-mail, list servers, electronic bulletin board or news systems, and electronic meeting systems. It can help organizations leverage their existing information infrastructures, improve overall productivity, and cope with diverse pressures and changing technologies.

Clarence Ellis and colleagues define groupware as "computer-based systems that support groups of people engaged in a common task (or goal) and that provide an interface to a shared environment."¹ Jay Nunamaker Jr. and colleagues² emphasize that groupware can also radically change the dynamic of group interactions by improving communication, structuring and focusing problem-solving processes, and establishing and maintaining an alignment between personal and group goals.

Electronic meeting systems

An electronic meeting system is a well-known example of groupware. An EMS consists of hardware, specialized software, and facilitation methods and techniques for solving group problems. It typically includes up to 30 networked PCs or workstations, special software that lets people enter comments and manipulate shared data, and some way to support the electronic projection of shared data. Software for general-purpose group problem solving consists of a set of programs that can be mixed and matched to suit the characteristics of both the problem and the participant group.

Research and experience with EMSs have demonstrated that electronic meeting support can improve a meeting's productivity. More people typically attend the meetings, less time is needed to gather information, and the group resolves the question or issue

the vector weights for all the nodes in its defined neighborhood. It computes the new weights using a simple error-correction function. The algorithm repeatedly presents the input data to the mapping layer starting with a large neighborhood radius and continuing with a gradually reduced radius. The algorithm converges when the radius becomes zero.

VISUALIZATION

The user can display results as either regions—in the default region mode—or as nodes.

Region view

Figure 1 shows a sample map for a GroupSystems session in which meeting participants answered “What issues are important to society in general, and the management of organizations in particular, and what are their implications for management, education, and research?”

As the figure shows, categories that occur more frequently occupy larger regions, such as faculty and management. Comments (documents) that discuss the same category are grouped within a region—in this case, education, knowledge, teaching, students, and curriculum.

Each region is marked with its name (the topic) and the number of documents—or in this case, comments—assigned to it. The color reflects the time of the comments. The scale progresses from cold to hot: dark blue to light blue to white to orange, and finally to rose. Thus, in this example, users can see at a glance that the discussion progressed from topics like schools and organizations to faculty and education, to change and leadership, to managers and technologies, and finally to accounting.

To determine which color a region should have, the tool averages and normalizes the time tags of comments in that region, compares the result to the results for other regions, and assigns colors to all the regions accordingly.

more quickly. Groups using an EMS have generated more unique alternatives for creative tasks and better decisions related to intellectual tasks than non-EMS-supported groups.²

EMSs enhance the group problem-solving process by allowing parallel, electronic discussions of complex problems. The electronic discussions are optionally anonymous and automatically recorded (a form of organizational memory), as opposed to a more exploratory “freethinking” process. Such systems identify the specific types of information needed for problem resolution. EMS users begin by gathering the information appropriate for the goal. Stages of this gathering process are divergence (idea generation), convergence (idea organization), and consensus checking (idea ranking or voting).

Lotus Notes

Electronic meeting systems are considered to be either same-time,

same-place or same-time, different-place groupware; that is, they support a focused, group problem-solving effort that is typically facilitated at one location or connected via networks and videoconferencing mechanisms. The meetings typically last hours or days in a retreat-style environment. Lotus Notes is a good example of different-time, different-place groupware; that is, it supports the electronic discussion of topics or projects over a longer period (possibly weeks).³ Lotus Notes is a group information manager that lets people be more effective in collecting, organizing, and sharing information across local area networks, wide area networks, and dial-up lines. Teams of people can integrate their knowledge, work processes, and applications across multiple computing platforms.

The fundamental concept in Notes is a database, a repository of information structured as a collection of documents or notes that users can view and organize in different ways. Unlike relational

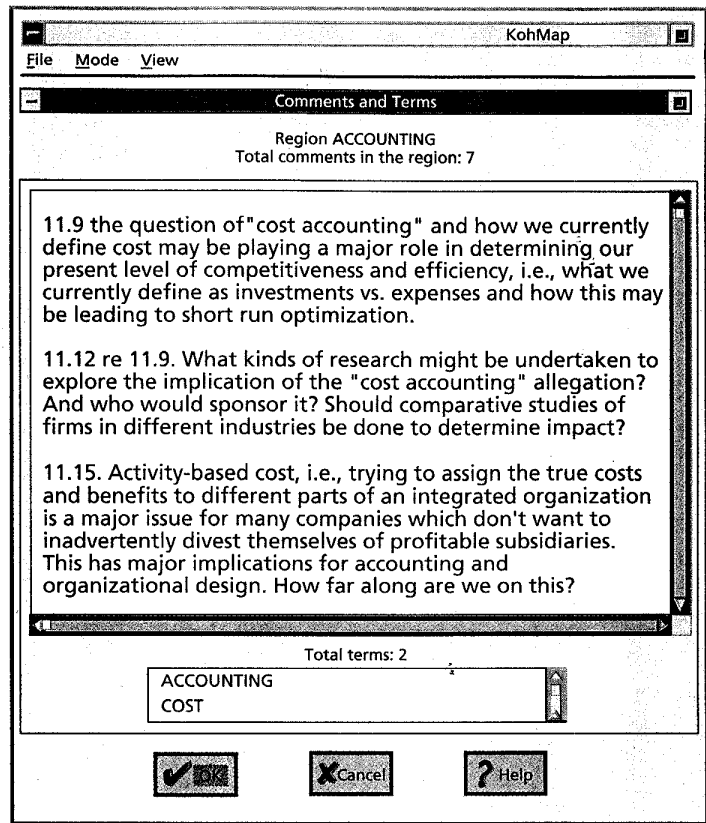


Figure 2. Documents assigned to the Accounting region in Figure 1. When the user clicks on Accounting, this dialog box appears.

When the user clicks on a region, the tool responds with a dialog box containing all relevant information about the region. For example, Figure 2 presents a dialog box for the accounting region in Figure 1. The upper part of the box contains the region’s title and assigned comments. The top text box contains the actual text associated with the topic, which users can examine for content and relevance. Usually, each document contains either a region title, which is the winning term described earlier, or terms assigned to that region. In Figure 2, the lower text window lists the terms relevant to accounting: “accounting” and “cost.” The number of terms assigned

tion, the concepts must be presented as building blocks only with subsequent levels involved with more theory.

A dialog box similar to that in Figure 2 lets users display the comments within a particular node position.

Miscellaneous category

The screens in both the region (Figure 1) and node (Figure 3) views contain a "file" pull-down menu with several options, including "miscellaneous." This category contains documents that do not have any of the assigned terms. Because all automatic cluster analysis techniques, such as the self-organizing map algorithm, are concerned with the frequency patterns of comments, comments that contain unusual ideas or typos are not assigned a region. However, we have found that, although this category can contain a lot of trash, there is also the occasional treasure. Figure 4 shows the miscellaneous comments from the sample session. Comments 3.8 and 3.10 are obviously not useful. However, comments 3.2 and 6.2 (and even to some degree 8.2) contain creative ideas and may provide insights into other aspects of the group's communication.

BENCHMARKING AND USER EVALUATION

To determine the efficiency of our prototype tool, we performed a benchmarking and subsequent user evaluation experiment. Table 1 presents the results of the benchmarking experiment. We began by conducting an extensive parameter test on eight archival GroupSystems files. We then generated results by altering values for six parameters: number of training iteration cycles, learning rate for the training phase, number of fine-tuning iteration cycles, learning rate for the fine-tuning phase, input vector size, and document frequency cutoff. We next tested eight GroupSystems files. The smallest was 11,891 bytes (64 comments); the largest, 125,472 bytes (737 comments). For each vector size and each file tested, we recorded the number of regions generated, the number of miscellaneous comments, and the processing time. We benchmarked the files on identical 66-MHz, Intel 486-class computers.

From these results, we identified several trends:

- *Number of regions.* When the input vector size increased (more terms and concepts to categorize), the number of regions also increased. However, the total number of regions generated was close to 30 in most cases.
- *Number of miscellaneous comments.* Not surprisingly, when the vector size increased, there were fewer miscellaneous comments. Thus, having a large vector size appears to be important to avoid creating too many miscellaneous documents.

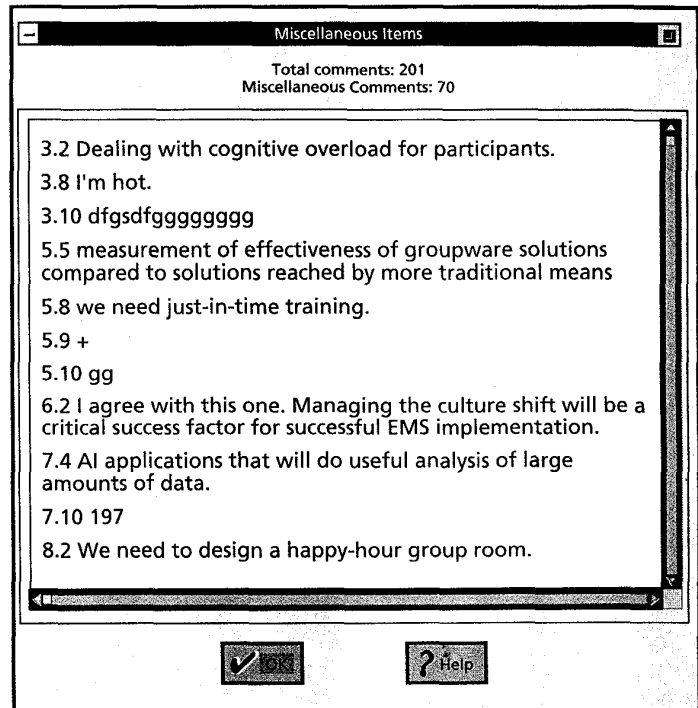


Figure 4. Display of miscellaneous comments. These are stray comments that don't fit into any region because the terms are used only once. However, as comments 3.2 and 6.2 illustrate, they may contain useful ideas.

- *Processing time.* The processing time increased with the file size and input vector size. However, even for the largest file, which had an input vector size of 100 terms, the tool finished in less than 13 minutes. Compared with the time-consuming (often several hours) and frustrating group convergence process, the tool's strawman results appear promising.

After the benchmarking, we conducted a test of how our tool compared with results generated by a manual analysis of meeting data using an archival GroupSystems file from an actual electronic brainstorming session. We chose the domain of collaborative systems because it was familiar to all eight facilitator subjects. During the electronic brainstorming process, an expert facilitator browsed the participants' comments and created a set of topics to categorize comments pertaining to the question, "What are the most important information technology problems with respect to collaborative systems to be solved over the next five years?"

Figure 5 shows the topic lists generated by the expert facilitator, who took about 40 minutes, and by our prototype tool, which took six minutes.

We asked each facilitator to compare and edit the two lists by adding new topics or removing and changing old ones. After they completed their changes, we computed the recall and precision levels of the two lists.⁷ *Recall* is the proportion of all relevant topics captured in the two lists. *Precision* is the proportion of the suggested topics—either by the human facilitator or our tool—that was considered relevant. Table 2 shows the results of this computation.

Each subject began with the 20 items from the expert

Table 1. Results of a benchmarking experiment to analyze eight GroupSystems files.

No. of comments	File size	Vector size	No. of regions	No. of miscellaneous comments	CPU time (sec.)
64	11,891 bytes	25	17	11	164
		45	29	9	280
		100	29	5	567
144	26,145 bytes	25	22	23	184
		45	32	15	287
		100	25	15	571
149	27,683 bytes	25	22	41	178
		45	25	33	285
		100	30	24	560
201	21,697 bytes	25	19	70	194
		45	33	46	300
		100	35	37	568
325	76,617 bytes	25	22	74	215
		45	30	55	321
		100	31	21	653
335	54,792 bytes	25	25	67	214
		45	26	51	304
		100	34	27	570
341	51,015 bytes	25	23	88	214
		45	28	61	307
		100	36	33	661
737	125,472 bytes	25	23	150	313
		45	30	92	446
		100	36	48	760

facilitator (Figure 5a) and the 27 items from the tool (Figure 5b) and added or deleted items that were deemed missing or irrelevant, respectively. Each subject's target number is the result of these additions and deletions.

In general, the tool's list was less precise (55 versus 81 percent), and the difference was statistically significant (a five percent confidence level). In contrast, the recall levels of the two lists (89 versus 81 percent) were not statistically different. We have found, however, that the low precision level does not preclude the tool's use in collaborative computing. Facilitators expressed a strong

interest in using the system-generated list as a strawman. They were generally dissatisfied with the time required and the cognitive demands of manual meeting convergence. Indeed, a joint system-user convergence process appears to be very feasible.

Although results from tests of our prototype tool are encouraging, more detailed experiments with a broader list of criteria are needed to exhaustively test the tool's efficiency and user reception. Such criteria might include letting users adjust the stop-word list (terms not to be indexed) to accommo-

Table 2. Comparison of output from human meeting facilitators (left) and prototype tool (right).

Subject	Target	Facilitator			Subject	Target	Prototype tool		
		Relevant	Recall (%)	Precision (%)			Relevant	Recall (%)	Precision (%)
1	15	14	93	70	1	11	9	82	33
2	32	14	44	70	2	32	11	34	41
3	10	10	100	50	3	10	7	70	26
4	22	20	91	100	4	20	18	90	67
5	19	18	100	90	5	22	21	95	78
6	17	16	94	80	6	14	13	93	48
7	19	18	95	90	7	24	24	100	89
8	22	20	91	100	8	19	16	84	59
Average			89	81	Average			81	55

1. video / projection
2. network / bandwidth
3. multimedia / hypertext / multimedia
4. group memory / project memory / repository
5. voice
6. culture / style
7. language
8. standards
9. distributed / distance issues / distance / different place
10. facilitation
11. research methodologies
12. cost / money
13. team
14. reward
15. integration
16. social / societal / society
17. performance
18. virtual
19. education / train / learn / teach
20. human / people / user / individual / inter-personal

(a)

1. technology
2. tools
3. meeting
4. support
5. collaborative
6. facilitator
7. matter
8. systems
9. people / issues / communication
10. application / integration / groupware
11. recognition
12. cultural
13. video
14. hardware
15. networks
16. dealing
17. users / group / reward
18. collaboration / seamless
19. memory / tool
20. training
21. virtual
22. environment
23. notes
24. information / ability
25. distributed
26. standard
27. bandwidth

(b)

Figure 5. Topic lists generated by (a) a human facilitator and (b) prototype tool. The human facilitator took 40 minutes to generate this list. The tool took six minutes to generate its list.

date terminology too general for a specific knowledge domain and varying the weighting scheme, for example, using a term frequency value in the vector instead of just a 1. That is, if a term appears three times, the weight would be 3. Other criteria might include accommodating user preferences for assigning documents to unique regions or multiple related regions. We plan to conduct systematic studies of user interaction and visualization effects in the near future.

Our current work involves adopting a similar approach for categorizing and analyzing Internet home pages² and archival and corrective databases. We are also comparing small-map multilayered results with large-map single-layered results to determine the utility of the map's hierarchical representation. We are also parallelizing several versions of the self-organizing map algorithm on HP/Convex Exemplar supercomputers and plan to introduce fisheye views and fractals to aid in map browsing. We expect such techniques to accommodate maps with several thousand output nodes. Finally, we are developing a Java-based version of the self-organizing map algorithm for interactive animation and a VRML-based version for 3D presentation and space navigation. ♦

Acknowledgments

We thank the anonymous *Computer* reviewers and contributing editor Nancy Talbert for their many helpful comments and suggestions in producing this article.

This ongoing project is supported mainly by the Digital Library Initiative, grants NSF/ARPA/NASA IRI9411318 and NSF/CISE: IRI9525790; by the AT&T Foundation Special Purpose Grant; and by the

US Army Corp of Engineers research contract DACA39-92-K-0042-P00003.

References

1. C. Upson et al., "The Application Visualization System: A Computational Environment for Scientific Visualization," *IEEE Computer Graphics & Applications*, July 1989, pp. 30-42.
2. H. Chen, C. Schuffels, and R. Orwig, "Internet Categorization and Search: A Machine Learning Approach," *J. Visual Communications and Image Representation*, Vol. 7, No. 1, pp. 88-102.
3. T. Kohonen, *Self-Organizing Maps*, Springer-Verlag, Berlin, 1995.
4. R. Miikkulainen, *Subsymbolic Natural Language Processing: An Integrated Model of Scripts, Lexicon, and Memory*, MIT Press, Cambridge, Mass., 1993.
5. X. Lin, D. Doergel, and G. Marchionini, "A Self-Organizing Semantic Map for Information Retrieval," *Proc. Int'l Conf. Research and Development in Information Retrieval*, ACM Press, New York, 1991.
6. T. Honkela et al., "Newsgroup Exploration with WEB-SOM Method and Browsing Interface," Tech. Report A32, Helsinki Univ. of Technology, Helsinki, Finland, 1996.
7. H. Chen et al., "Automatic Concept Classification of Text from Electronic Meetings," *Comm. ACM*, Vol. 37, No. 10, pp. 56-73.

Hsinchun Chen is professor of management information systems at the University of Arizona and head of the UA/MIS Artificial Intelligence Group. He is also principal investigator of the Illinois Digital Library Initiative project funded by NSF, ARPA, and NASA.

His research interests are in semantic retrieval, search algorithms, knowledge discovery, and collaborative computing. He is also a visiting senior research scientist at the National Center for Supercomputing Applications. Chen received a PhD in information systems from New York University.

Jay Nunamaker Jr. is regents professor of management information systems and computer science at the University of Arizona and director of the university's Center for the Management of Information. He has published more than 200 papers and seven books dealing with group decision support systems, the automation of systems development, databases, expert systems, systems analysis and design, and strategic planning. His work has also appeared in Forbes, Fortune, Business Week, The Wall Street Journal, The New York Times, and the Los Angeles Times. Nunamaker received a BS in industrial management from Carnegie Mellon University, a BS in mechanical engineering from the University of Pittsburgh, an MS in industrial engineering from the University of Pittsburgh, and a PhD in operations research and systems engineering from the Case Institute of Technology.

Richard Orwig is an assistant professor of management information systems at Washington State University. Previously he was a research scientist at the University of Arizona's Center for the Management of Information. His interests include object-oriented requirements engineering and how to facilitate group problem-solving sessions using GroupSystems software. Orwig received a BS in mathematics and a BA in philosophy from the University of Illinois and an MBA and a PhD in management information systems from the University of Arizona.

Olga Titkova is a software engineer at Intel. Her current interests include project management, technical support, and customer interaction activities. Titkova received an MS in mathematics from the Kiev University and an MS in management information sciences from the University of Arizona.

Contact Chen at Dept. of Management Information Systems, University of Arizona, Tucson, AZ 85721; hchen@bpa.arizona.edu or Orwig at Dept. of Management Information Systems, Washington State University, Vancouver, WA 98686; orwig@wsu.edu.

you@computer.org FREE!

All IEEE Computer Society members can obtain a free, portable email
alias@computer.org.

Select your own user name and initiate your account.

The address you choose is yours for as long as you are a member.

If you change jobs or Internet service providers, just update your information
with us, and the society automatically forwards all your mail.

**Sign up today at
<http://computer.org>**

