Association for Information Systems
AIS Electronic Library (AISeL)

AMCIS 2005 Proceedings

Americas Conference on Information Systems (AMCIS)

2005

Testing an Integrity-Checking DSS Component for Organizational Memory Building

Dianne J. Hall
Auburn University, halldia@auburn.edu

Yi Maggie Guo
University of Michigan - Dearborn, magyiguo@umd.umich.edu

Testing an Integrity-Checking DSS Component for Organizational Memory Building

Dianne J. Hall
Auburn University
dhall@auburn.edu

Yi Maggie Guo
University of Michigan - Dearborn
magyiguo@umd.umich.edu

ABSTRACT

Organizational Memory (OM) has become a critical component of organizations attempting to maintain a competitive advantage. The ability to retrieve accurate information, and interpret it from different perspectives, will both facilitate efficient problem solving and support efficient organizational learning. This research examines the effect of an information-building and decision aid. The results indicate that the system positively affects OM by promoting user-entered knowledge bases with more accurate statements. However, the system did not significantly improve the ability of the users to solve a problem. These findings have important implications for designing information systems to support and expand OM while increasing the organization's capacity to learn and solve problems.

KEYWORDS

Decision Support, Organizational Memory, Integrity Checking

INTRODUCTION

Organizational memory (OM) may be a critical component in an organization's search for competitive advantage (e. g., Bordestsky and Mark, 2000; Tuomi, 2000; Wijnhoven, 1999). Data and information that is stored by an organization is of limited use if it lacks integrity or if it becomes stagnant. Thus, a modern organization's needs go beyond simple data and information storage and retrieval to building OM that contains accurate information with a minimum of redundancy. Errors and redundancy in OM can limit its usefulness, thus reducing learning possibilities that otherwise may have served to increase the breadth of OM and encourage creative thinking that leads to organizational innovation.

The ability to facilitate verified OM by promoting user input that is correct, not redundant, and structured for search and retrieval must be considered as investigation into developing information building and decision aids continues. Without a good foundation, later research into less structured environments may not be well facilitated. Thus, this research examines whether an information building and decision aid with an integrity-checking component will promote users to enter more correct facts and fewer incorrect facts into OM. Additionally, it examines whether such a system will reduce the amount of redundancy in a user's input. Secondarily, this research examines the question of whether providing such an error checking component will make a difference in the user's ability to arrive at correct decisions. The system described in this paper encourages users to solve a problem and, in doing so, populate OM with items useful for subsequent problem solvers. The findings indicate that use of the system results in more accurate data input but not a significant difference in problem solving when compared to the control system.

BUILDING ORGANIZATIONAL MEMORY

Organizational Memory (OM) represents the hard and soft facts embedded throughout the organization. Soft facts (e.g., tacit knowledge, expertise, etc.) reside in the collective memory of individuals; hard facts are generally stored either electronically or on paper. OM that is both accurate and broad-reaching has been linked to many organizational benefits. It has been shown to increase organizational effectiveness (Jennex, Olfman, Panthawi and Park, 1998), improve core competence, reduce transaction costs, increase efficiency and effectiveness in decision-making, and increase organizational learning (e.g., Cross and Baird, 2000; Hedberg, 1981; Stein, 1995; Stein and Zwass, 1995). It has also be shown to increase effectiveness of other organizational support systems such as decision-support systems, expert systems, and knowledge management systems (King, Marks Jr. and McCoy, 2002; Tiwana, 2001).

Proper design, testing, and evaluation of OM systems are critical before they can be engaged to provide the support an organization requires. Fortunately, the act of developing information systems specifically to support OM often leads to explicating information within a storage system and providing communication across the organization through system requirement definition, thereby helping integrate the knowledge through the organization.

At the most fundamental level, OM must provide support for accurate, timely, and easily retrieved information. To address this issue, the primary research question in this study is whether a system providing information building guidance promotes quality of OM content. Among the ways to increase information accuracy is to elicit the help of the user in ensuring that information has been carefully considered and structured prior to being entered. We conceptualized and tested an information building aid that controls how and what artifacts are placed into OM by developing an integrity-checking component in a small decision support system. Accuracy should be controlled by examining user input against known "truths" or data, and warning the user if inconsistencies are uncovered. This system feature is not only necessary for OM, but also may act as decisional guidance for the user. Such guidance may be in the form of informative guidance or suggestive guidance (Parikh, Fazlollahi and Verma, 2001; Silver, 1991). Our system provides informative guidance by providing the user with relevant information in response to either a query or an attempted entry. Suggestive guidance, which provides the user with potential actions, is intentionally not included as such would confound our secondary analysis as to whether an integrity-checking OM building component of a decision support system will also improve decision accuracy.

The system tested in this research acts much like the mnemonic layer of an OM information system (OMIS) as described by Stein and Zwass (1995). These functions include knowledge acquisition (user input), retention (storage of that input in a database), search within and retrieval from the database of both user entries and previously existing entries, and maintenance (integrity). While our system can only maintain integrity when the user responds accordingly, it could easily be changed to mandate integrity by prohibiting entry of information that now generates only a warning. Alternatively, simple heuristics can be used to clean the database on a regular basis to eliminate redundancy or error. In fact, because an OMIS can be misused and, as a result, become dominated by a specific context or item, improper use has been defined to include the introduction of incorrect (or incomplete) information. Thus, our component is important in this respect. Practically, it is expected to impact decision-making by controlling the facts that get stored, which should result in lowered transaction costs for similar decision domains (e.g., Walsh and Ungson, 1991).

Information overload can occur with useful information; however, when the knowledge base also contains inaccurate or redundant information for a given context, the effect of information overload is exacerbated. Studies indicate that information overload contributes to incomplete problem formulation and less desirable problem solutions (Hwang and Lin, 1999; Kivetz and Simonson, 2000). The system tested here reduces information overload by considering incoming information in light of existing information to determine whether a user's potential additional to OM is incorrect or a duplication of known information. The system issues a series of warnings when information is found to be incorrect or redundant, prompting the user to reconsider the input. The implication of wrong information is obvious; the implication of redundant information is less so. If there is too much information in OM, users may not sift through the information to find what is relevant, possibly foregoing a well-informed choice. It is posited that receiving warnings against potentially inaccurate or redundant information causes a user to more carefully consider the appropriateness of information being placed in OM.

Capable integrity-checking components embedded in DSS should be designed to encourage users to enter only correct information and to forego entering that which is incorrect or redundant. We posit that such a system will increase the overall accuracy of OM by reducing the amount of error and redundancy it contains. Thus, we hypothesize:

> **H1:** Participants using the system with an integrity-checking component will enter fewer erroneous items in OM than participants who use the system without an integrity-checking component.

> **H2:** Participants using the system with an integrity-checking component will enter fewer redundant items in OM than participants who use the system without an integrity-checking component.

Because the proposed system is expected to decrease inaccuracy and redundancy in OM by warning the user about both, we expect that a user will quickly understand that entries made without generating a warning, by default, are correct and will facilitate solving the puzzle. Therefore, they will pay more attention to feedback on their entry attempts. While the integrity-checking component of the system is not designed to increase number of right entries per se, a natural outcome of this anticipated behavior is that the number of correct entries will increase. Thus, we hypothesize:

> **H3:** Participants using the system with an integrity-checking component will enter more items that are correct into OM than participants who use the system without an integrity-checking component.

If the user is entering correct information, then the user also is aware of the correctness of their assumptions regarding the problem's solution. Can we then assume that those individuals will arrive at the correct solution more often than those who use the system without the integrity-checking component? Because both systems used in the study should promote proper decisions, yet one of the systems essentially gives the user feedback as to the correctness of their assumptions, we hypothesize:

> **H4:** Participants using the system with an integrity-checking component will be more successful in solving the problem than participants who use the system without an integrity-checking component.

THE STUDY

Participants were 285 students enrolled in undergraduate Information Systems courses at a major southwestern university. Students were appropriate choices as participants because they were less likely to have had previous experience with an information building system or a decision aid (which may bias their response to the experimental system). Also, the problem to be solved with the decision aid did not require any specific knowledge of business procedures or situations, so experience in a business setting was not required.

All participants solved similar tasks on an information building and decision support system (described below). The tasks were logic puzzles, asking the user to pair first and last names of four individuals, and the individual with an object purchased. There were clues help participants solve the puzzle. This type of intellective task is appropriate for initial testing of a system and its design, particularly one that provides problem-solving support in a structured environment. It also allowed for variance in the solution so that a user might approach the correct answer (correctly identifying all twelve relationships) and not be penalized for a singular right or wrong answer. Keeping all tasks similar while changing specific information between tasks allowed the system design to remain identical between treatment groups, thus eliminating the possibility of task/user interaction. The puzzle was first pilot tested on paper to determine whether it contained an appropriate number of clues with which to solve the puzzle, but was complex enough that the participants would need to interact with both the query and input features of the system to complete the task of solving the puzzle. The test indicated that the two versions of the puzzle that contained thirteen clues provided participants with enough information to solve the puzzle in a reasonable amount of time for a lab setting. The clues were entered by researchers into OM as initial items.

There were two versions of the system. A basic system was used as a control, and an extended (treatment) system with the integrity-checking component added. Both systems contained a series of input forms that were designed to allow the participant to query the existing knowledge base for information that could be used to solve the puzzle. Both systems were coded using Visual Basic as a front-end and an individual Access database to collect the results of each participant's session. The first screen of the system was the puzzle face itself (Appendix 1). Similar in style to logic puzzles found in common puzzle books, the screen allowed the participant to visualize the problem on a grid-like form and choose action buttons as necessary. Available actions included Help, Query, Add to Knowledge Base (i.e., OM), View Input, or Exit. Clicking on Help brought up a pop-up message box explaining how to use the system. Choosing Query brought up the Query screen (Appendix 2), which allowed the participant to search the knowledge base for existing information regarding the puzzle. In this screen, two of three keywords must be entered. For instance, a participant may choose "Laura" and "Ames" to query OM for information on that combination, or "Laura" and "Dress" to query for that combination. Clicking on Add to Knowledge Base allowed the participant to enter information into OM using a structured format for defining keywords. The View Input button brought up the Edit/Delete screen allowed the participant to view the information they had input as well as make corrections or delete the information. Participants were prevented from editing or deleting the initial set of items (the clues). The Exit button was clicked only after the participant had completed the puzzle to their satisfaction. Clicking on Exit saved the entries made and the puzzle solution, printed a copy of the puzzle grid, and exited the program. In addition, all interaction between the system and the participant (queries, entries, warnings/information received, etc.) was recorded in a transaction log in the database.

The treatment system was identical to the control system with the addition of the integrity-checking component that examined user-entered information for items that were contrary to the clues or that duplicated either the clues or items the user had previously entered; in this case, the user was warned by a message and given the option to retract their entry. In no case was the user required to refrain from entering information about which they had been warned. To address the issue of using similar words, like "buy" and "purchase", a strict structure, rather than natural sentences, was used. That is, only keywords (last name, first name, and objects) were stored. On the puzzle screen, a check in the row of "Amy" and column of "Ring" indicates that "Amy bought a ring." An item with values of "Amy" and "Ring" will be inserted into OM once user clicks Add to Knowledge Base. When querying, user enters the values of keywords and all the items meeting the requirements will be displayed on the screen.

The experiment was a between subject design with participants randomly assigned to either a control or treatment group, with two versions of the task nested within each group. Two similar tasks were used because the procedures called for multiple participants to be run simultaneously in a crowded lab and it was desired that participants not be able to easily see the work of another participant on the same task.

The experiment was conducted in three phases: orientation, training, and task. Prior to orientation, participants were randomly assigned to one of four possible treatments groups (Control/Puzzle 1, Control/Puzzle 2, Treatment/Puzzle 1, and Treatment/Puzzle 2). During the orientation phase the administrator explained the steps the participants would take during the experiment, the task, and in general terms described the software system that would be used. The participants were then trained on a mock system that did not include a specific task or OM, but included numerous explanations of the various forms they would encounter. The participants were led by the administrator through each of the screens and were encouraged to click on buttons and text boxes in order to understand the workings of each form and their relationship to the task in general. Each form was programmed with a number of pop-up message boxes that explained the area of the form that the participant was viewing. Participants were encouraged to solve the puzzle and to add information to OM such that another individual would be able to query the system to quickly find the solution to the puzzle. Participants were not told when or how much information to enter.

RESULTS

Information on the number and type of user-entered items was gathered by examining user-entered items in OM. Because of the strict keyword structure of the system design, no equivocality existed regarding coding. The number of incorrect items entered was determining by counting items that were directly contradictory to the initial clues. For instance, because "Andy did not buy a belt" is a correct item, "Andy bought a belt" would be incorrect. Duplicates were determined in the same manner – items that that appeared more than once in the knowledge base, whether based on the initial clues or user-entered information, were counted as being duplicates. A correct entry provided both new and correct information. For instance, an item such as "Betty bought a ring" would be useful if no information existed in OM (either as initial clues or user input) that was a duplicate of or in contradiction to it. The correctness of the user's solution was determined by examining the results of the puzzle which had been entered into a separate table. A correct solution to the puzzle yielded twelve correct items, while failure to solve the puzzle at all resulted in zero correct items.

Preliminary Analysis

The participants were 46% female, 44% percent male, and 96% ranged in age from 19 to 24 (four percent of the participants were older than 24). Seventy one percent of the participants were either freshmen or sophomores, and 63% percent were enrolled in the College of Business. Of the 285 participants who began the experiment, 22 experienced catastrophic machine failure (i.e., crashes) from which recovery was impossible. Thus, those participants were eliminated from the final analysis leaving 263 usable participants. No other participants experienced malfunctions of any kind.

The data was examined for outliers using the boxplots for each of the variables. Twelve participants were identified as outliers; however, visual inspection of the data indicated that there were no entry errors, indicating that these outliers are likely the result of individual differences of the participants and therefore remain in the analysis. In addition, transaction logs were examined to determine whether any of the treatment group participants failed to receive integrity-checking feedback. All participants received at least one so all remained in the analysis. There are 131 participants in the control group and 132 in the treatment group for the analysis below.

The variables of interest were then checked for normalcy and homogeneity of variance. All variables except the correctness of the solution were found to have similar variance with the Levene test; however, they demonstrated significant non-normal patterns with both Kolmogorov-Smirnov and Shapiro-Wilk statistics. Both ANOVA and Kruskal-Wallis ANOVA were conducted; the results were similar. Because one variable of interest was normally distributed, and ANOVA is robust to non-normality when the sample is sufficiently large, ANOVA is used in the results below. Statistical checking for possible effects of demographic differences (age, classification, major), experimental sessions, machines, and puzzle version was conducted but no effect was found. Thus, any resulting differences can reasonably be attributed to the tested systems, which is the sole interest for the remainder of the statistical analysis.

Main Analysis

Hypothesis 1 suggested that a user of the treatment system would create a knowledge base (i.e., expand OM) with fewer incorrect items than users of the control group would. ANOVA results suggest that system has no effect for the number of incorrect items in the knowledge base (F=.008). Hypothesis 1 is not supported. Both the control group and the treatment group, on average, entered 1.60 incorrect items. This indicates that the integrity-checking component designed to decrease incorrect entries had no effect on the users who continued to enter those items.

Analysis of Hypothesis 2, however, indicates that the integrity-checking component designed to warn against duplicate entries did make a difference in duplicate entries in the knowledge bases of the treatment group. Our results indicate a significant effect (F=56.49), indicating that 17.5% of the variance in duplicate entries can be attributed to the system. On average, individuals who used the integrity-checking systems entered one duplicate item in the knowledge base, whereas those who used control system, on average, entered three and a half items. Hypothesis 2 is supported.

Hypothesis 3 continues to look at items in the resulting knowledge bases, positing that interaction with the integrity-checking component will lead to a higher number of correct items being entered than in the control system. Results indicate that the treatment group entered more correct items (4.94) than the control group (3.75), a significant difference (F=6.94); however, system does little to explain the variance (2.2%). Hypothesis 3 is supported.

Hypothesis 4 posits that, again because of interaction with the integrity-checking component, participants in the treatment group are more likely to approach the correct puzzle solution than those in the control group. Our analysis reveals that there is no effect of system on reaching a solution to the problem (F=.966). Out of a score of twelve (perfect response), both groups averaged between seven and eight. Hypothesis 4 is not supported.

DISCUSSION

These results indicate that the inclusion of a specific integrity-checking component in an information building and decision aid produces some desired results. That is, a system that incorporates that component results in a knowledge base with fewer duplicate items. We believe that this is a direct result of the feedback inherent in the component; in all, users were not inclined to enter items they knew already existed in the knowledge base. Although both group could have checked for duplicates before entering their items and thus reduce redundancy on their own, it appears that individuals may work more on a "enter first, apologize later" basis without an explicit integrity-checking component. In this study, the control group entered significantly more duplicates than the treatment group, although all participants were directly encouraged to consider what the knowledge base should contain to make it easy for subsequent problem solvers. It is possible that the participants did not consider problems inherent with redundancy until those in the treatment group were specifically warned about it. Therefore, the treatment system with integrity-checking component served to limit redundancy.

What is of particular interest in the results is why the users heeded system warning regarding duplicates but did not appear to heed the warning regarding errors. To further analyze this question, we examined each of the groups taking into account only those individuals who had incorrectly solved the puzzle. This group should have had the most errors of all. Once again, however, we found no system effect. Looking at all groups, however, a trend emerged. People who solved the puzzle incorrectly had significantly more incorrect items than those who solved the puzzle correctly or those who hadn't completed the puzzle, regardless of system. In all three solution groups (solved correctly, solved incorrectly, or didn't solve) the treatment group participants had fewer incorrect entries. There was no interaction effect between system and the groups, and system still has no effect.

Following through on this analysis, we then looked at the number of correct entries. Like the results above, those who solved the puzzle correctly should have the highest number of correct entries. This is true once again – a significant difference in correct entries is seen for the correctly solved group than either of the other groups. Overall, the treatment group had more correct entries regardless of whether the participant solved the puzzle correctly or not. However, this analysis indicates that, when puzzle solution is taken into effect, there is a significant effect of both factors, and no interaction. Variance explained increases to 5.5% with this model. The addition of solution to the model made no difference in the duplicate entries results.

Our results support our hypotheses that an interactive integrity-checking system makes a significant different in the number of correct items and duplicate items in OM, but there was no statistical support for fewer incorrect items. Further investigation may be necessary to determine whether this unexpected outcome is the result of an effect size that is small enough to be indistinguishable with the current sample, or whether, in fact, there is a tendency among individuals to ignore information they believe is false and, as a result, propagate their errors. This may also explain why there was no significant difference in solution quality. On the other hand, this result has important implications for the design of information systems in general. It has established that simply informing a user that their assertion is incorrect may not change their behavior. Further, the decision aid, combined with the interactive integrity-checking component, did not make a difference in the final solution of the problem. Better solutions to both issues should be sought in the future.

The results of this study are limited in their generalizability by some facets of the research design. Student participants are often considered to be a limiting factor when discussing transfer of results to the organizational setting. Further, lab settings may be strong on "rigor" but generally must forego an element of "relevance" (Mason, 1988; Mason and Mitroff, 1973). However, using students at this stage of experimental process and the task is appropriate for our goals.

Most of the limitations in this study result from the need to narrow the scope to the fundamentals of the system in order to allow rigor during testing and because of the lack of cohesive empirical literature on the impact of DSS to OM. Another limitation of the study may be the strict structure of both the query and the input components. This limitation was necessary to support accurate measurement and identification of the redundant and incorrect assertions from the user, but may not represent some systems which may be more of a free-form nature. Again, future research may examine this issue, perhaps by using intelligent agent technology.

A final limitation to this research is the task. This type of intellective task is representative of more traditional DSS capabilities, which provides problem-solving support in a structured environment. For situations of less structure, however, a different task should be implemented. For instance, the system should be tested with a task that presents an unstructured problem with many potential perspectives and for which there is not only a selection of possible solutions, but also the possibility of no solution.

Beyond the future research alternatives that come from limitations of the current study, there are many avenues of future research available. An important line of research may be to more closely examine the behavioral effect of this system and attempt to identify how often and why individuals ignored the integrity-checking component's warnings and what psychological perceptions users holds for those warnings. A similar study with system log and screen monitoring software will enable us to take a closer look of this phenomenon with supplement data. Thus, a better solution to decrease the number of incorrect entries can be derived.

Another research line should examine the impact of integrity-checking components on decision-making and organizational learning should be considered. Beyond the obvious positive effect of reduced redundancy and increased accuracy, we can begin to investigate additional components that would appear to impact these tasks. For instance, an advanced support system, when embedded in a comprehensive support system, is capable of internal information discovery and proactive information acquisition. These components should lead to broader-reaching and more accurate OM from which timely decision-making and organizational learning will benefit.

CONCLUSION

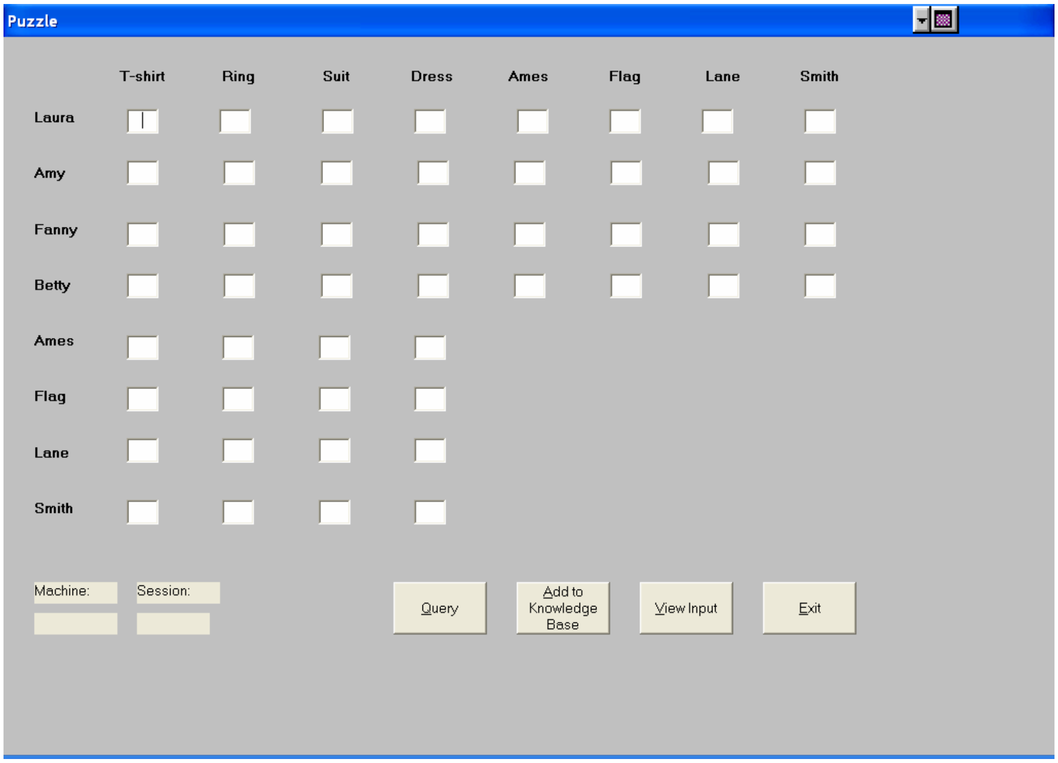
This study provides a small but significant step toward conceptualizing an integrity-checking component within a decision support system, with both research and practical implications. This research has shown that focusing an information building and decision aid on the integrity of OM will provide an organization with OM that contains more accurate items and fewer redundant items than a system that does not contain integrity-checking components. The practical implication is to include an integrity-checking component in to any decision support system development. Such a component may be crucial to the quality of organizational memory, particularly in a user-entered information building system. A simple warning can reduce duplicates; however a better way to reduce incorrect entries is still under investigation. This research contributes verification for a type of integrity-checking component. Organizations that adopt such a system can expect to reduce time spent on ineffective data mining by reducing the amount of redundancy in their OM. Further, the combination of a decision support system and feedback associated with an integrity-checking component may encourage increased user input, possibly providing organizations with increased creativity and innovative thinking, which will ultimately lead to better decisions and more effective organizational learning.

REFERENCES

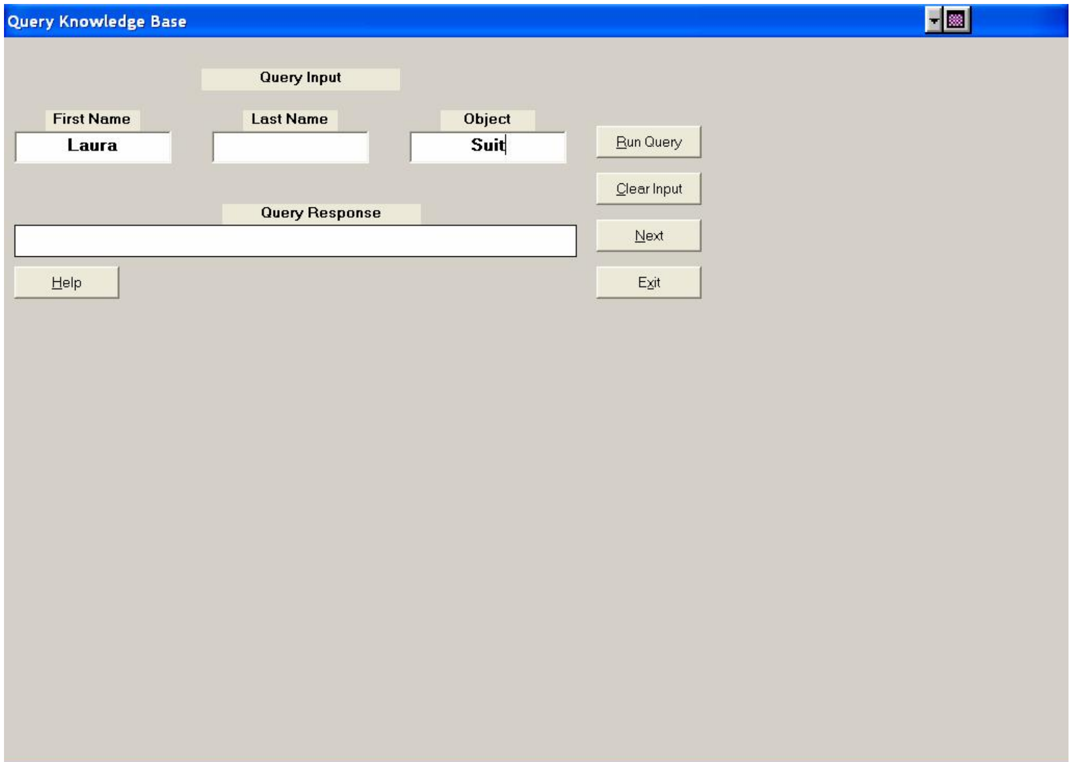
1. Bordestsky, A. and Mark, G. (2000) Memory-based Feedback Controls to Support Groupware Coordination, *Information Systems Research* 11, 4, 366-385.
2. Cross, R. and Baird, L. (2000) Technology is Not Enough: Improving Performance by Building Organizational Memory, *Sloan Management Review* Spring, 69-78.
3. Hedberg, B. (1981) How organizations learn and unlearn, in *Handbook of Organizational Design: Adapting Organizations to their Environments*, P. C. Nystrom and W. H. Starbuck (Eds.), Oxford University Press, New York, 3-27.
4. Hwang, M. I. and Lin, J. W. (1999) Information Dimension, Information Overload, and Decision Quality, *Journal of Information Science* 25, 3, 213-218.
5. Jennex, M., Olfman, L., Panthawi, P. and Park, Y. (1998) An Organizational Memory Information Systems Success Model: An Extension of DeLone and McLean's IS Success Model, *Proceedings of the Hawaii International Conference on Information Systems*, Maui, Hawaii, 157-165.
6. King, W., Marks Jr., P. and McCoy, S. (2002) The Most Important Issues in Knowledge Management, *Communications of the ACM* 45, 9, 93-97.

7. Kivetz, R. and Simonson, I. (2000) The Effects of Incomplete Information on Consumer Choice, *Journal of Marketing Research* 37, 4, 427-448.
8. Mason, R. O. (1988) Experimentation and Knowledge: A Pragmatic Perspective, *Knowledge* 10, 1, 3-24.
9. Mason, R. O. and Mitroff, I. (1973) A Program for Research on Management Information Systems, *Management Science* 19, 5, 475-487.
10. Parikh, M., Fazlollahi, B. and Verma, S. (2001) The Effectiveness of Decision Guidance: An Empirical Evaluation, *Decision Sciences* 32, 2, 303-331.
11. Silver, M. S. (1991) Decisional Guidance for Computer-Based Decision Support, *MIS Quarterly* 15, 1, 105-122.
12. Stein, E. W. (1995) Organizational Memory: Review of Concepts and Recommendations for Management, *International Journal of Information Management* 15, 1, 17-32.
13. Stein, E. W. and Zwass, V. (1995) Actualizing Organizational Memory with Information Systems, *Information Systems Research* 6, 2, 85-117.
14. Tiwana, A. (2001) The Essential Guide to Knowledge Management, Prentice-Hall, Englewood Cliffs, NJ.
15. Tuomi, I. (2000) Data is More than Knowledge: Implications of the Reversed Knowledge Hierarchy for Knowledge Management and Organizational Memory, *Journal of Management Information Systems* 16, 3, 103-117.
16. Walsh, J. P. and Ungson, G. R. (1991) Organizational Memory, *Academy of Management Review* 16, 1, 57-91.
17. Wijnhoven, F. (1999) Development Scenarios for Organizational Memory Information Systems, *Journal of Management Information Systems* 16, 1, 121-146.

Appendix 1. Puzzle Screen



Appendix 2. Query Screen



Appendix 3. Input Screen