AI and Skillful Systems in Reporting Informations: Study and Extensions

Naveen Kumar
Dept. of CSE, Teerthankar University, Moradabad, UP, India

Abstract

The purpose of this paper is to survey and extend the use of Artificial Intelligence and expert systems in accounting databases. The paper elicits a number of concerns often voiced about accounting databases. The use of Artificial Intelligence and expert system is investigated as a basis to mitigate those problems. The literature is surveyed and extended. Demons and objects are found to be useful devises to facilitate the organization, storage and application of intelligence for accounting database systems. Models for their use are presented.

Keywords

AI, Expert System, Accounting Databases

I. Introduction

Accounting Information Systems moved out of the arena of paper journals and ledgers and into computer-based formats with the advent of computers. Unfortunately, in many cases all that was done was to develop computerized systems that the computer used as a more efficient type of paper processor or calculator. Consequently, in many cases, accounting databases have become vast storehouses of limited information about specific accounting transactions. As a result, these systems do not meet the needs of decision makers. One approach to this problem is to integrate Artificial Intelligence (AI) into accounting databases to try to develop systems that mitigate the difficulties of traditional systems.

Although, accounting database theory has received substantial attention, little work has been done on the application of AI/ expert systems (ES) to accounting Requests for reprints should be sent to Daniel E. O'Leary, School of Business, University of Southern California, Los Angeles, CA 90089-1421. A survey of the literature of accounting applications suggests most of the previous AI/ES work in accounting has focused on auditing, with some work in managerial accounting and tax applications. Thus, it is critical to examine the problems in accounting database theory and investigate the extent to which AI/ES can mitigate those difficulties. Thus, the approach to this paper is to elicit some of those difficulties and then investigate that integration with three primary purposes. First, much of the literature on the use of AI/ES in accounting databases is reviewed to establish the current state of application. Second, other research in AI/ES (e.g., Herschberg, 1986, and Parsaye, 1989) is examined for its contribution to developing intelligent accounting databases. The emphasis in that part of the paper is on the use of demons and objects in accounting database systems to mitigate some of the problems elicited. Demons and objects are presented as means to organize, store, and apply the necessary intelligence in the systems. Third, additional problems and extensions to the use of AI/ ES in accounting databases are examined.

A. Difficulties with Existing Accounting Database Systems

Researchers have noted the following difficulties with current accounting database systems.

1. Accounting Information Not Meeting Needs of Decision Makers

Accounting researchers often have argued that conventional accounting systems do not meet the needs of their users. McCarthy (1982) noted that accounting databases do not include related non-accounting information. For example, productivity and reliability data are often too aggregated, use inappropriate coding schemes, and are not adequately integrated with the data needs of the rest of the firm.

2. Inability for Humans to Process or Understand What is Captured in the Computerized Accounting Databases

The ability of computer-based systems to ac-cumulate and store accounting information is now enormous. Large volumes of data compounded with decision time constraints have been found by researchers to lead decision makers to make suboptimal decisions (White, 1983). In addition, few accounting models have been offered that change the way to model and use the data. Traditional income statements, cash flow statements, and balance sheets still are the primary models used in summaries of accounting data. Thus, it is not just that decision-maker needs are not met; in some cases the users do not know how to use the avail-able data and in other cases, time limits their ability to use the available data.

3. A Focus on Numeric Data

The ability to process numeric (syntactic) data typically has been regarded as the strength of computerized systems. Consequently, systems have been designed with an emphasis on numeric data. However, this has led to the exclusion of much symbolic (semantic) data and models that process both numeric and text data, which can be useful in assessing important context & other variables associated accounting events, e.g., including information such as who initiated (processed, etc.) a transaction and the motivation of that person.

4. Interpretation of the Relationship Between Transactions to Yield Actual Events

With increasing computerization of manual paper-generating processes some of the benefits of having humans more involved has been lost. Humans used to be able to bring understanding and memory to the processing of accounting information. However, often there is little information in computerized accounting databases about how or if different transactions are related to the same event. For example, additional no accounting information about the specific causation of those transactions (and other context-oriented information) could be helpful in establishing such relationships.

5. Systems are Difficult to Use

Users either will not use systems that are not easy to use or will experience substantial costs in the use of those systems. Ease of use is likely to be a function of the interface with the system and the ease with which the underlying models (on which the system is based) are understood or congruent with decision makers. For example, databases with natural query language are likely to be regarded as easier to use, than systems where natural language is not available.

B. Contributions of Artificial Intelligence

AI/ES can have a substantial effect on accounting databases in mitigating some of these problems. ES technology suggests developing models that can assist the decision maker and focus on decision-maker information needs (e.g., Hayes-Roth et al., 1983). Computer-based systems, with AI can exploit the power of the computer and investigate substantial detail. Further, recent developments in AI/ES have stressed the integration of context and symbolic information. Some artificial intelligence tools can facilitate a broader understanding of the events captured by the accounting system. For example, symbolic knowledge can be used to determine that apparent disparate in-formation is related. Further, a simple trip to the library computer retrieval system will convince anyone that some database users are better at information retrieval than on others. Capturing those models, say as an expert system, could facilitate database use for many other users. More than just numeric information is required to understand the environment of the firm. Integrating intelligent systems with accounting databases can assist (either with the decision maker or independent of the decision maker) in the investigation of large volumes of data with or without the direct participation of the decision maker. Thus, systems can analyze the data and assist the users in understanding or interpreting transactions to determine what ac-counting events are captured by the system. Natural language interfaces can facilitate the use of most systems. In addition, the cognitive processes and knowledge structures are also a concern of Al. In the case of accounting database systems, this means studying and building models of the way that, say, expert accounting database users make use of an accounting database. Such models could facilitate use of the systems since they are congruent with the way the expert user views the world.

C. Outline of This Paper

This paper proceeds as follows. Section 1 identifies problems with traditional accounting databases and suggests that AI be used to investigate and extend ac-counting database systems. Section II, briefly discusses some background terminology of so-called "events" accounting databases (an approach that has been proposed as a framework for the viewing accounting data). This section concludes that the events approach is tied to a classic decision support system approach, and that it is desirable to advance beyond that approach to one where expertise is built into the system. Section III, summarizes the previous research in artificially intelligent accounting numeric and text database systems. Section IV, investigates the introduction of natural language systems into accounting systems. This approach indicates the need to understand the structure of accounting language and the corresponding knowledge structures that underlie that language. Section V, discusses extensions to the previous research based on demons, and Section VI, investigates object-oriented computer programming. In particular, the issues addressed in those sections include defining and ascertaining the existence of events, rather than transactions; integrating symbolic and numeric Information in accounting databases; and capturing additional relevant context information about the firm through such databases. Sections V and VI, present methods to organize, store and apply intelligence to mitigate the difficulties identified. Section VII, discusses some additional extensions, to accounting databases.

Finally, Section 8 provides a brief summary of the paper. Throughout, the focus of this paper is on the domain of accounting databases and the use of Artificial Intelligence in those databases. Although most of the focus is on the solution and structure of problems, that approach is consistent with research to date on accounting database systems.

II. Background-Events Accounting

Recent accounting database theory (e.g., McCarthy, 1979, 1982) has focused on an "events" approach. Generally speaking, events accounting database is aimed at capturing "events" that affect a firm. The events theory approach to accounting databases probably is the most accepted theoretical approach to the design and development of accounting databases. (McCarthy's [1979, 1982] implementation of events theory is based on the ER approach of Chen [1976, 1980]).Sorter (1969) observed that accountants seemed to have two different perspectives on accounting information: value and events. The value perspective suggests that the choice of accounting data for a database is normative-information for accounting databases is chosen to assist the decision maker. Because the choice of some data leads to the elimination of other data, an underlying theory was assumed for or with the decision maker. The events approach suggested that accounting is concerned with providing data that is not tied to particular database designers, but instead could be used in a number of decision situations.

Events theorists have suggested that Event theorist have suggested that if accounting data were available on all accounting events then there would be no need for formalized aggregations of the data or models of the firm, such as traditional financial statements. For example, as noted by Sorter (1969), "Instead of producing input values for unknown and perhaps unknowable decision models directly, ac-counting provides information about relevant economic events that allows individual users to generate their own input values for their own decision models" (p. 13).Further, Sorter (1969) notes that "In a subsequent manuscript, I intend to speculate on the type of accounting reports appropriate to this approach" (p. 15). If users of an events accounting system wanted in-formation, then they could search the database and formulate the appropriate models to analysis or summarize the data. The database would not limit the user by imposing models on the data. Unfortunately, the events approach currently suffers from some of the same limitations as traditional ac-counting database systems. First, even systems touted as being events accounting systems are aimed primarily at accounting information and accounting events (McCarthy, 1979, 1982). Thus, certain functional in-formation (such as production) is eliminated from the view of the decision maker by the lack of their inclusion in the database. As a result, such systems still do not meet all of the information needs of decision makers. Second, a system that depends on each individual user's ability to ferret out important data, decide how to use that data, etc., neglects the impact of time and human limitations, such as those discussed by Simon (1981) on users. Third, because of its accounting focus, the events approach is aimed at capturing numeric accounting data. As a result, symbolic information that may be quite useful in defining an event is not captured. Fourth, in many cases, what constitutes an event is not clear. The classic example is the case of the purchase of a $1,000 piece of equipment by a manager limited to purchases of $500. One way to circumvent the process is to make two $500 purchases-yet in systems with much human intervention, these kinds of purchases are difficult to get through the system. is clearly the $1,000 purchase. Recording the event in multiple $500 transactions or in different time periods

A. Relationship to Decision Support Systems

The events approach is consistent with a “database dominated” decision support system approach that was gaining prominence at the time of Sorter’s (1969) paper. The database was used to support decision making, not to limit or make decisions for the user. The user, it was reasoned, could analyze the data with any of a variety of statistical or analytical tools. At this time, technology was not sophisticated enough to play a proactive role in decision processing. As a result, the notion of events and decision support systems of this type did not recognize the apparent potential performance differences between different users-some users are more expert than others.

B. Relationship to Artificial Intelligence Systems

Unfortunately, the current structure of the events approach suffers from being limited by the technology in which it was conceived: at the time of its development making process did not exist. The development of AI and ES provides an opportunity to build intelligence or expertise into the database in order to assist users. Such models could assist users by sorting through large quantities of data without the user’s direct participation, assist the decision maker under time constraints, suggest alternative models to evaluate or search for data, etc. In addition, the development of AI would suggest that rather than just numeric data, symbolic information also be captured to additionally characterize the transaction. Further, it suggests the use of natural language processes and expert models be developed in the systems to facilitate interaction of the user with the system. Unfortunately, use of AI/ES in accounting database systems is not straightforward. Thus, this paper addresses the extraction, organization, storage, and application of intelligence to accounting databases.

III. Previous Research: Accounting Database Systems and AI

Database theory has been substantially integrated into the development of accounting databases. This development has taken two distinct formats: numeric and text. Recent research has extended these efforts using AI and ES.

A. Numeric Databases

Bailey, Hun, Stansifer, and Whinston (1988) summarized accounting database models based on the taxonomy in Brodie, Mylopoulos, and Schmidt (1984), using two broad categories: classical data models and semantic data models. Classical data models took three formats. The first form was a hierarchical approach to structuring accounting information and was explored by Colantoni, Manes, and Whinston (1971) and Lieberman and Whinston (1975). The relational database approach was brought into accounting by Everest and Weber (1977), while the design of a multiple-dimensioned accounting system using a network approach was investigated by Haseman and Whinston (1976).

The development of semantic database models of accounting information brought some of the under-lying semantic structure to accounting databases. Se-mantic data models were first introduced into accounting by McCarthy (1979, 1982) using Chen’s (1976) entity-relationship model to develop the entity-relationship view of accounting and the REA (resources, events, and agents). More recent efforts, such as those of Bailey et al. (1988), also fall into this category. There has been limited research in the investigation of the use of semantic databases in real-world settings. Gal and McCarthy (1986) discussed the procedures necessary to maintain a relational accounting database and retrieve information to meet various needs. Weber (1986) studied the order entry modules of 12 wholesale distribution packages to determine the use and effectiveness of the REA model.

B. Text Databases

Accounting text databases were developed concurrently with developments in numeric accounting databases. The first text databases of interest to accountants were NARS and LEXIS. These databases contain publicly available accounting and legal information about selected companies, such as news articles and annual re-ports. More recently, EDGAR (electronic data gathering and retrieving system) has allowed companies to file their required disclosures with the SEC (Security and Exchange Commission) in an electronic format.

C. Artificial Intelligence and Expert Systems in Numeric Accounting Databases

Although both accounting numeric and accounting text databases exist, this review suggests that there has been little or no discussion of their integration into a single database structure. In addition, there has been only limited work in interfacing -research in numeric accounting databases (e.g., REA databases) with decision systems. Denna and McCarthy (1987) developed a prototype decision support version of the theoretical model presented in McCarthy (1979, 1982). However, that system contained little knowledge or intelligence not contained in the database schema. In a related study, Storey and Goldstein (1990) developed an expert system to elicit user views during logical database design. The system elicits information requirements through a dialogue with the user and re-solves inconsistencies, ambiguities, and redundancies.

D. Artificial Intelligence and Expert Systems in Text-Based Databases

It often is necessary to search text databases to find information about particular accounting pronouncements or about the characteristics of particular firms. Recent developments in AI have led to the development of “smart” computer-assisted search through these databases. At least three prototype systems have embedded search intelligence within the text databases. Arthur Anderson (1985a, 1985b) and Mui and McCarthy (1987) developed two systems to interface with EDGAR. One system, ELOISE (Arthur Anderson, 1985a) was designed to search through an ASCII version of data from EDGAR in order to find documents that related to anti-takeover provisions. Another system, FSA, was designed to search through various disclosures (also represented in ASCII), including text, in order to calculate various financial ratios. These systems employed the work of De Jong (1979) to structure the understanding of text.

O’Leary (1988) developed an intelligent system to search through an ASCII version of LEXIS and NEXIS data to overcome some of the difficulties of using text systems in accounting databases. The system was de-signed to include “search concepts,“ similar to those used in ELOISE (rather than simple “key word” searches), “found concepts” (that test the text found in order to see if it matches what was being searched for), “expert search plans” (that employ domain knowledge normally attributed to librarians) and, “remembering and forgetting” (to assist in subsequent search efforts-also characterized as learning and unlearning).

The system discussed in O’Leary (1988) was based on the knowledge acquired from a librarian specializing in information

 ISSN : 0976-8491 (Online) | ISSN : 2229-4333 (Print)

retrieval. Thus, the system was an attempt to mimic that librarian, at a prototype level, in order to build expertise into database search.

IV. Previous Research: Natural Language In Accounting Systems

Natural language systems continue to be an area of development in AI (Allen, 1987). Some of the most powerful and creative approaches developed in AI have been devoted to examining natural language systems. Although generic natural language interfaces have been developed for databases, those systems seemingly have not exploited the structure in accounting language (Tanaka, 1982).

The role, importance, and impact of natural language interfaces in databases in general is not clear (Sethi, 1987). Thus, it is not surprising that there has been only limited work on interfacing natural language-based systems with accounting databases to facilitate use of the database. Research in natural language is critical since such front-ends on databases facilitate ease of use. In addition the study of natural language in accounting systems is critical since language is one of the only maps that we have to the underlying knowledge structures.

A. Developing a Chart of Accounts

O'Leary and Munakata (1988, 1989) developed an approach to the processing of a given set of accounts, using a natural language description of those accounts and financial information about those accounts, in order to develop a chart of accounts for an accounting system. The system was developed to take into account appropriate intelligent behavior, such as minimizing disclosure of sensitive information and maximizing the inclusion of appropriate levels of aggregation for decision making. These systems exploited existing ac-counting theory in the development of the system and the knowledge structures of a management consultant. Later tests of those systems found that the systems could produce charts of accounts similar to human analysts. The system was also better at developing charts of accounts than no experienced users. Thus, the systems could interact with accounting information for the structuring of an accounting database by employing knowledge structures that apparently were similar to human system developers.

B. Selecting Natural Language Understanding in Accounting

Subsequent research concerned with processing natural language inquiries in accounting database systems (O'Leary and Kandelin, 1991) investigated the power of a very limited vocabulary in terms of representing particular accounting events. This research demonstrated that by exploiting the structure of accounting language, the expression of accounting concepts could be summarized in very parsimonious forms. For ex-ample, the "purchase" of goods generally is expressed in a natural language format in a limited number of ways (e.g., "purchased" or "bought"). Further, once it is ascertained that an event is a particular kind of accounting event it is easy to search through the remaining communication of that event to determine characteristics of that event. For example, the fact that an event is a "purchase" implies the existence of a vendor, a price, a quantity, etc. Other assumptions can be made in some systems, such as the existence of a purchase order, a purchasing recommender, and other directly linked activities. In addition, the assumptions that can be made "explode" back from the transaction. For ex-ample, in the case of a purchase, there is a production need and marketing support for the resulting product. Information on each of these could then be captured for the resulting database.

By making initial assumptions about the context, concepts such as these can be achieved using a limited understanding of natural language. Throughout, much of the parsimony is achieved because of the reliance on the exceptions of the underlying model on which the data are based or context from which the data are derived. These investigations of the language used by accountants Suggest that a few concepts (represented in the system Design as objects and in the system implementation as frames) can be used to characterize a broad range of what is recognized as accounting trans-actions and language. They also suggest knowledge structures used by accountants to summarize their worlds. This study was based primarily on normative accounting theory and was an attempt to attain the understanding of a beginning accountant.

V. The Application of "Demons" to Accounting Databases

The previous research on AI/ES accounting databases has addressed some important problems. However, it has neglected one of the problems at the very base of events accounting theory. Since events theory is at the base of most contemporary accounting database theory, this is critical. In addition, as seen above, events ac-counting generally is tied to a decision support approach, rather than an AI/ES approach. Thus, in part, the purpose of this and the next section are to investigate methods to identify events. These sections also attempt to bring the events approach into AI/ES framework. Both demons and objects are seen as approaches that provide the ability to organize, store, and apply the necessary intelligence to make accounting databases intelligent.

A. Demons

Demons, with origins in both databases and Al, offer much potential to accounting database systems (Rich, 1983; Winston, 1984). Their use can affect some of the difficulties elicited in the first section of the paper, particularly the identification of events. Demons are a useful programming tool designed to provide various updates to the databases as various events occur. As noted by Winston (1977), "Demons are subroutines that are called automatically by specified database additions and removals . . . they keep watch over what goes in and what comes out and activate themselves when something goes by that they like" (p. 379). Winston (1977) suggests that there are two reasons for using demons:

Demon's behavior is activated by data received, not because some program requested that they be activated. "Demons add knowledge to a system with-out specification of where it will be used. . . . Like competent assistants they do not need to be told when to act."

Since demons provide an "independent" function, they are not part of the main program. "Demons encapsulate bookkeeping operations that otherwise litter programs . . . Programs become more read-able . . ." (p. 380)

As a result, demons offer an important device for integrating AI into accounting databases. Demons provide intelligence by monitoring data in the system and activating themselves only in appropriate situations.

B. Intrusion-Detection Systems

Typically, demons have been developed to watch over different patterns of activity. As a result, demons have been employed as the basis of systems for auditing and security of computer-based systems. Such systems are called intrusion-detection systems since they are normally designed to detect unusual activity, such as intrusions into a system. These systems establish expectations and then monitor data to determine if expectations are met.

Denning (1987) and Tener (1988) have developed intrusion-detection systems to protect computer systems and databases, respectively. These systems make use of expectations of the user. For example, statistics of a user might include, when the system is used, what printers are used, etc. Thus, when that same user signs on at an unusual time, at an unusual location, and decides to print on a printer never used before, the system may take additional steps to ensure that user is who they say they are. Vasarhelyi, Halper, & Fritz (1989) presented a system described as providing the "continuous audit of online systems." Using various metrics, the system monitors transactions and compares the monitored information with the expected information to deter-mine the existence of unusual transactions. As noted by the authors the system ". . . allows for the capture and impounding of auditor expertise both into the measurement analytics as well as into system probes" (P. 1).

C. Event System Uses of Demons

As discussed above, one of the problems in events-based systems is that events may not be defined appropriately, particularly if the system takes the data as it is given. For example, an event may be defined by more than a single transaction, either in a single period or different periods. Unless the system is intelligent enough to see that difference, it may not function effectively.

1. Relating Two or More Transactions to Establish Underlying Events

Typically, firms employ spending limits (authorization levels) on employees in order to decentralize responsibility, yet maintain control over costs and employee behavior. As noted in an example discussed earlier in the paper, a common ploy to circumvent those limits is to break an expenditure that exceeds those limits into two or more pieces that do not exceed those limits. Arrangements are made with the vendor and multiple bills (transactions) are received for the same event. Under current accounting systems, unless something is noticed by the human caretakers of the accounting system, each of the multiple bills will be treated as different transactions by the system-even though they both relate to the same events.

A demon can examine transactions to us if they relate to another transaction in that time period. This can be done by using heuristics similar to those that might be used by human investigators, such as examining each transaction with a given vendor, or each transaction authorized by a given manager to deter-mine "relatedness" of transactions.

2. Relating Information from Different Time Periods

A similar set of issues is faced by transactions that could occur in different time periods, yet are still related to the same transaction. For example, when breaking the purchase of a piece of equipment into two transactions because of a ceiling on purchase price, those two transactions may be put into different periods. Demons can search out such transactions, by relating transactions in different period, with vendor, transaction type, or authorization source.

3. Implementation of Demons to Identify Events

If all transactions were compared to one another then in firms where there are literally billions of transactions, this approach could require infeasible amounts of re-sources. If, however, demons employed human investigator's heuristics then the approach could become computational feasible.

An initial study of a manual accounting system yielded some additional heuristics for matching trans-actions to events, including the following:

- Unless there is evidence to the contrary, assume that transactions are events.
- If told that transactions are related to the same event, then assume that they are related to that event unless there is evidence to the contrary.
- Work to establish evidence that transactions are related to other transactions.
- Examine transactions near the spending limit for their relation to other transactions.
- For transactions that appear to be matched with other
- Transactions, disregard previously identified completed events, unless there is some reason to reopen the file (e.g., suspicious set of transactions).
- Gather information on the "who" and "what" associated with the transactions (who initiated the transaction and what was the transaction for). Apparently, some individuals are more likely to do this than others and apparently some cases of multiple expenditures are more likely associated with a single event than others (education/travel/software).
- Pay particular attention to transactions related to departments that have broken events into multiple transactions in the past.
- Supplement existing accounting numeric records with "notes" summarizing unusual aspects of trans-actions and events.

These and other heuristics can be part of a demon-based system, where the demon's job is to identify groups of transactions that could be parts of the same events. Such an approach would employ these heuristics to cut down the potential size of the combinatorial space.

VI. Applications of Object-Oriented Programming

Objects are a way of viewing the world. Objects can be e.g., things or activities. Objects were used as a means of capturing "concepts" in the above section on natural language. Object-oriented programming languages (OOPLA) are software that allows the user to focus on and characterize particular entities or objects. Typically, everything in these languages is treated as an object. Examples of OOPLAs (Stefik & Bobrow, 1986) include Actor (Whitewater Group, 1987) and Small-talk.

A. Objects

Objects are a unique type of programming approach, allowing the combination of data and knowledge. As noted by Stefik and Bobrow (1986), "objects are entities that combine the properties of procedures and data since they perform computations and save local state" (p. 41). In object-oriented programming all the activity arises from messages either being sent to objects or by objects. Objects can respond to messages much as individuals would respond to them. Each object can use a different set of procedures to process messages. In addition, objects employ a hierarchical structure, so that any object lower in the hierarchy maintains the properties of any object above it in that hierarchy.

However, given the availability of technology such as Actor (Whitewater Group, 1987) we can expect the development of other systems. The one system that has received probably the most attention is discussed in Apte et al. (1988), Kastner (1986),

ISSN : 0976-8491 (Online) | ISSN : 2229-4333 (Print)

and Mays (1987). The FAME (financial and marketing expertise) system, developed at IBM, employs objects and rules in a complex knowledge structure. In that system, an "event" is treated as an object. The event is the decision suggested by the system, ranging from one hierarchical level of "outright purchase" to lease to lease with option to buy. In addition, objects employ a hierarchical structure, so that any object lower in the hierarchy maintains the properties of any object above it in that hierarchy.

B. Previous Accounting Object-Oriented Systems

There have been few examples of accounting or financial-based object-oriented systems developed either as research prototypes or as actual function systems. However, given the availability of technology such as Actor (Whitewater Group, 1987) we can expect the development of other systems. The one system that has received probably the most attention is discussed in Apte et al. (1988), Kastner (1986), and Mays (1987).
The FAME (financial and marketing expertise) system, developed at IBM, employs objects and rules in a complex knowledge structure. In that system, an "event" is treated as an object. The event is the decision suggested by the system, ranging from one hierarchical level of "outright purchase" to lease to lease with option to buy.

C. Objects and Accounting Databases: Conceptual Design

The definition of an event in the system described here is different than the definition of events for accounting systems. The event in FAME is an outcome. In ac-counting database systems the event is something that has happened, is happening, or is about to happen. The critical aspect in an event accounting system is characterizing what defines an event, what are the relevant information views of the event, what information is needed to characterize that event, and how that in-formation is best captured (e.g., numeric).
Thus, the notion of objects and events are consistent with each other. In this system, as events and trans-actions occur they provide input to the objects of the system. Information on virtually all feasible types of information would flow into the system. The objects in the system would then be responsible for choosing the information they need.
Thus, one view of an object-oriented system de-signed to be an abstractor of information for database purposes could be to represent each of the demands for different views of information as a set of hierarchically related objects. For example, accounting information needs would be under the control of an ac-counting set of objects, production information from the event would be under the control of a production set of objects, etc. In each of those functional areas an REA or arbitrary relational database approach would be developed. This approach would allow a broader definition of an event than just its accounting perspective, and may include other characteristics from other disciplines that better capture the nature of the event. For example, a purchase from a vendor may not only result because of the quality and price of the product, but also because it is viewed as a marketing effort to that firm for the sale of its own products. Such reciprocity often occurs in business settings, but is seldom captured in data-bases.
By sending messages back and forth, objects can be used to model the reciprocal relationships between different views. For example, in accounting there is now a focus on integration of production measures of quality into accounting systems (Johnson & Kaplan, 1987). Such concerns could be captured in part by the process of message sending. Accounting objects concerned with quality information could gather their own from the messages sent to the system or could rely on other production objects to gather the information. Not all objects would be concerned with each trans-action or event. For example, in the set of accounting objects, there could be both accounts payable and ac-counts receivable objects. Clearly, payable and receivable objects rarely would be concerned with the same transactions or events. However, there are situations where there are overlapping needs for information. Those needs can be established by sending and receiving messages.
In addition, a set of objects could be concerned solely with the determination of what is an event. Such objects would be concerned with relating different messages to each other. Objects can be constructed so that they search for different types of information, much as different humans in organizations search for information to meet their needs. Then the objects would extract structure data in a suitable manner. Event information provided to the system would consist of a wide range of information, including economic transactions, information on "Acts of God," such as earthquakes, since they could impact accounting variances in prices, marketing sales, finance . . . etc. Other context establishing data, such as scanned documents (e.g., purchase orders, other documents or written communications), voice messages, electronic mail, etc., could be linked as part of a text-based data-base under any of the functional areas. Thus, objects can be used to allow the system to capture more symbolic and semantic context information than would be possible with a traditional database.
Although the discussion has been aimed primarily at the extraction and storage of information, such a system could also be designed to process requests for information. Objects would be responsible for knowing about the existence of data and different formats of data. Requests for information would then be directed to appropriate objects that contained knowledge about the database and requests, such as the system in O'Leary (1988).

VII. Additional Database Issues

There are a number of other related emerging issues in the area of AI/ES and databases that can directly affect accounting databases and establish additional research issues.

A. Smart Convergence of "Old Files Into New"'

As firms and governments begin to try to use data files established before documentation standards were well-established, they are finding, in some cases, that the exact format or content of some files is unknown. In order to decipher what is on the files, firms and governments search out humans who were affiliated with the original projects, if they can be found. Alternatively, if no such persons exist or if there is little memory of the files then alternative approaches must be developed.
One approach is the development of expert systems to assist in the process of converting old undocumented files into files that can be used. Thus, expert systems are beginning to be proposed to process the data to find out format or content of the databases. Although such expert systems have not been reported extensively in the literature, there are some basic statements that I would like to thank K. Bimson for bringing this problem to my attention can be made about them. First, they employ knowledge about databases in general for both target and source. Thus, the systems have substantial knowledge about relational databases and their construction, such as that captured in Storey and Goldstein (1990). Second, the systems can attempt to include "local" expertise to assist in determining relationships between fields, etc. Some employees may regularly work with portions of the database.

Knowledge of particular fields can be built into the system. In addition, partial documentation or knowledge can be ascertained from examining the databases or programs that use the databases. Such knowledge also could be built into an expert system. Third, file conversion expert systems can employ heuristics that a human expert would use to determine the underlying structure. For example, such a heuristic might be "if the data range is 32 to 48 else 0 then the likely field is hours worked per week." Fourth, once any information is found or thought to be found in a database, that information can be used to infer the existence of other information. For example, the existence of hours worked in a week suggests that other likely fields are employee number, pay rate, etc. The user's past interaction with the system. From a similar perspective, systems could be developed to learn what presentation methods the decision maker uses, what data are used etc., and then provide that data to the user in anticipation of user needs. Roth and Mattis (1990) have addressed some of these issues from a general perspective, but accounting database presentation of such information has not yet is discussed.

B. Smart Restructuring the Organization of the Database

Currently, periodically the accounting database of an organization is redesigned and restructured to meet the changing needs of the organization. Recently however, AI researchers, including Dejong (1979) and Kolodner (1980) report that systems need to have the capability to modify the structure of the knowledge used by the system. As a result, it is tempting to suggest that an adaptive accounting database system would have this same capability. Such a system could periodically re-view the use and demands for information, and expected relationships, as the basis of an effort to restructure itself.

However, there could be substantial concern with such a system. For example, there would be security issues, continuity issues, and archival issues that would need to be addressed. Further research is underway in the investigation of this notion.

C. Smart User Interfaces

User interfaces go beyond the need for natural language approaches discussed in Section 5. Interfaces also include graphics and other forms of presentation. There has been a substantial amount of research in the past into the presentation of information to users (Reneau & Grabski, 1987). Since individual users may use different forms of data presentation to analyze results, it could prove useful for the system to anticipate which data presentation form The user would use (e.g., charts or bar graphs). This determination could be based on the user's past interaction with the system. From a similar perspective, systems could be developed to learn what presentation methods the decision maker uses, what data are used etc., and then provide that data to the user in anticipation of user needs. Roth and Mattis (1990) have addressed some of these issues from a general perspective, but accounting database presentation of such information has not yet is discussed.

D. Models to Process Database Information

As noted above, researchers have suggested that new accounting models be developed to meet the needs of the events perspective. However, there have been few new accounting models added to the portfolio of approaches to analyze accounting database information. One approach to finding new models may be the traditional expert systems approach-find someone who is an expert in analyzing the data and then build a system that mimics some of their problem-solving or database-search behavior.

VIII. Summary

In the first section of this paper it was noted that ac-counting database systems had been criticized for the following.

1. not meeting the needs of decision makers;
2. having so much information that humans could not process or understand what was in the accounting database;
3. It focusing on numeric data;
4. not understanding or interpreting events; and
5. Difficult to use.

In the review of the events-based approach to database systems, it was found that contemporary approaches still faced some of the same criticisms.

A survey of some recent uses of Al, ES, and natural language in accounting database systems found that some of those limitations had been addressed. How-ever, there still has been limited work in determining how intelligence is organized, stored, and applied in the context of accounting database systems. This paper provides some approaches to mitigate the critisms and yet, provides a basis for the integration of integrating intelligence into accounting database systems. Demons (to link events) and objects (to focus on and integrate other types of data and views of data) were investigated to further mitigate some of those limitations. Further, some additional extensions were discussed based on recently elicited problems in the area of accounting database systems, including converting old files to new, integrating smart user interfaces that adapt to the way the user solves the problem, and finding new models of expert use of accounting data-bases.

References

[1] Allen, J., Apte, C., Griesmer, J., Hong, S.J., Karnaugh, M., Kastner,"Natural language understanding", Menlo Park, CA: The Benjamin/Cummings Publishing Company, 1987.

[2] J., Laker, M., Mays, E.,"Experiences with object centered modeling of financial marketing. Unpublished paper presented at the 1988 First International Symposium on Expert Systems in Business, Finance and Accounting, University of Southern California, Los Angeles, CA, 1988.

[3] Arthur Anderson,"Financial statement analyzer (Final Report)". Unpublished report, 1985a.

[4] Arthur Anderson,"ELOISE Project Report", Unpublished report, 1985b.

[5] Bailey, A., Hun, K., Stansifer, R., Whinston, A.,"A formal alogorithmic model compatible with conceptual modeling in accounting information systems", Unpublished working paper, University of Texas, Austin, TX, 1988.

[6] Brodie, M., Mylopoulos, J., Schmidt, J.,"On conceptual models", New York: Springer-Verlag, 1984.

[7] Chen, P.,"The entity-relationship model-Towards a unified view. ACM Transactions on Database Systems, March, 9-36.

[8] Colantoni, C., Manes, R., Whinston, A.,"A unified approach to the theory of accounting and information systems", The Ac-counting Review, 47(1), pp. 90-102, 1971.

[9] Dejong,"Skimming stories in real time: An experiment in integrated understanding", Unpublished doctoral dissertation, Yale University, New Haven, CT, 1979.

[10] Denna, E., McCarthy, W.,"An events accounting foundation for DSS implementation", In C. Holsapple & A. Whinston (Eds.), Decision support systems: Theory and application. New York: Springer-Verlag, 1987.

[11] Denning, D.,"An intrusion detection model", IEEE Transactions on Software Engineering, 13(2), pp. 222-232, 1987.
[12] Everest, G., Weber, R.,"A relational approach to accounting models", The Accounting Review, LII(2), pp. 340-359, 1977.
[13] Gal, G., McCarthy, W.,"Operation of a relational accounting system", Advances in Accounting, 3, pp. 83-112, 1986.
[14] Haseman, W., Whinston, A.,"Design of a multidimensional accounting system", The Accounting Review, LI(1), pp. 65-79, 1976.
[15] Hayes-Roth, F., Waterman, D., Lenat, D.,"Building expert systems. Reading", MA: Addison-Wesley, 1983.
[16] Johnson, H., Kaplan, R.,"Relevance lost", Boston, MA: Harvard Business School Press, 1987.
[17] Kastner, J., Apte, C., Griesmer, J., Hong, S.J., Karnaugh, M., Mays, E., Tozawa, Y.,"A knowledge based consultant for financial marketing. AI Magazine", 7(5), pp. 71-81, 1986.
[18] Kerschberg, L.,"Expert database systems", Proceedings from the First International Workshop", Menlo Park, CA: Benjamin/ Cummings Company, Inc, 1986.
[19] Kolodner, J.,"Retrieval and organizational strategies in conceptual memory: A computer model", Unpublished doctoral dissertation, Yale University, New Haven, CT. Kolodner, J., & Reisbeck, C. (1986). Experience, memory and reasoning. Hillsdale, NJ: Lawrence Erlbaum Associates.
[20] Lieberman, A., & Whinston, A.,"The structuring of an events accounting information system", The Accounting Review, L(2), pp. 246-258, 1975.
[21] Mays, E., Apte, C., Griesmer, J., Kastner, J.,"Organizing knowledge in a complex financial domain", IEEE Expert, 2(3). McCarthy, W. (1979). An entity relationship view of accounting models. The Accounting Review, LIV(4), pp. 667-686, 1987.
[22] McCarthy, W.,"Construction and use of integrated accounting systems with entity-relationship modeling", In P. Chen (Ed.), En-tity-relationship approach to systems analysis and design. Amsterdam: North Holland, 1980.
[23] McCarthy, W.,"The REA accounting model: A generalized framework for accounting in a shared data environment", The Accounting Review, LVII(3), pp. 554-578, 1982.
[24] Mui, C., McCarthy, W.,"FSA: Applying AI techniques to the familiarization phase of decision making", IEEE Expert, 2(3), pp. 33-41, 1987.
[25] O'Leary, D.,"A prototype expert system for information retrieval", Proceedings of the National Meeting of the Decision Sciences Institute. Las Vegas, NV, 1988.
[26] O'Leary, D., Kandelin, N.,"Accountant: A domain dependent accounting language processing system", Artificial Intelligence in Accounting and Finance. Amsterdam: North-Holland (Forth-coming), 1991.
[27] O'Leary, D., Munakata, T.,"Developing consolidated financial statements using an expert system", In E. Turban & P. Watkins (Eds.), Applied Expert Systems. Amsterdam: North Holland, 1988.
[28] O'Leary, D., Munakata, T.,"An accounting prototype expert system", In M. Vasarhelyi (Ed.), Artificial Intelligence in Accounting and Auditing. New York: Marcus Wiener, 1989.
[29] Parsaye, K.,"Expert database systems: Object-oriented, deductive hypermedia technology", New York: John Wiley and Sons, 1989.
[30] Reneau, H., Grabski, S.,"A review of research in computer-human interaction and individual differences within a model for research in accounting information systems", The Journal of In-formation Systems, 2(1), pp. 33-53. Rich, E. (1983). Artificial Intelligence. New York: McGraw-Hill, 1987.
[31] Roth, S., Mattis, J.,"Automatic graphics presentation for production and operations management", Proceedings of the Fourth International Conference of Expert Systems in Production and Operations Management (pp. 493-509). University of South Carolina, Columbia, SC, 1990.
[32] Sethi, V.,"Towards understanding the role, importance and impact of natural language interfaces to databases", International Conference on Information Systems, pp. 176-186, 1987.
[33] Simon, H.,"The sciences of the artificial", Cambridge, MA: MIT Press, 1981.
[34] Sorter, G.,"An "Event" approach to basic accounting theory. The Accounting Review, 44(1), pp. 12-19, 1969.
[35] Stefik, M., Bobrow, D.,"Object-oriented programming: Themes and variations", AI Magazine, pp. 40-62, 1986.
[36] Storey, V., Goldstein, R.,"An expert view creation system for database design. Expert Systems Review, 2(3), PP. 17-43, 1990.
[37] Tanaka, S.,"The structure of accounting language", Tokyo, Japan: Chuo University Press, 1982.
[38] Tener, W.,"Expert systems for computer security", Expert Systems Review, 1(2), pp. 3-6, 1988.
[39] Vasarhelyi, M., Halper, M., Fritz, R.,"The Continuous Audit of Online Systems", Unpublished paper presented at the 1989 International Symposium on Expert Systems in Business, Finance and Accounting, University of Southern California.
[40] Weber, R.,"Data models research in accounting: An evaluation of wholesale distribution software", The Accounting Review, LXI(3), pp. 498-518, 1986.
[41] Winston, P.,"Artificial intelligence. Reading", MA: Addison-Wesley, 1977.
[42] Winston, P.,"Artificial intelligence (2nd ed.)", Reading, MA: Addison-Wesley, 1984.
[43] Whitewater Group,"Actor language manual", Evanston, IL: The Whitewater Group, 1987.
[44] White, C.a,"Aggregation in internal accounting reports and decision making: A field experiment approach", Paper presented at the American Accounting Association Meeting, New Orleans, LA, 1983.

Naveen Kumar completed his Bachelor of Technology (Computer Science and technology) degree from Radha Govind Engineering College, Meerut (UPTU, Lucknow) in 2008, pursuing Master of Technology (Computer Science and technology) degree from Teerthanker Mahaveer University, Moradabad. He completed his Microsoft certification professional in Asp.net in 2011. Before M.Tech he had worked with a company "Ablaze Infotech Pvt Ltd, Delhi" as Software Developer One Year, in 2009-10. After that he switched toward further study to share and explore his work experiences.