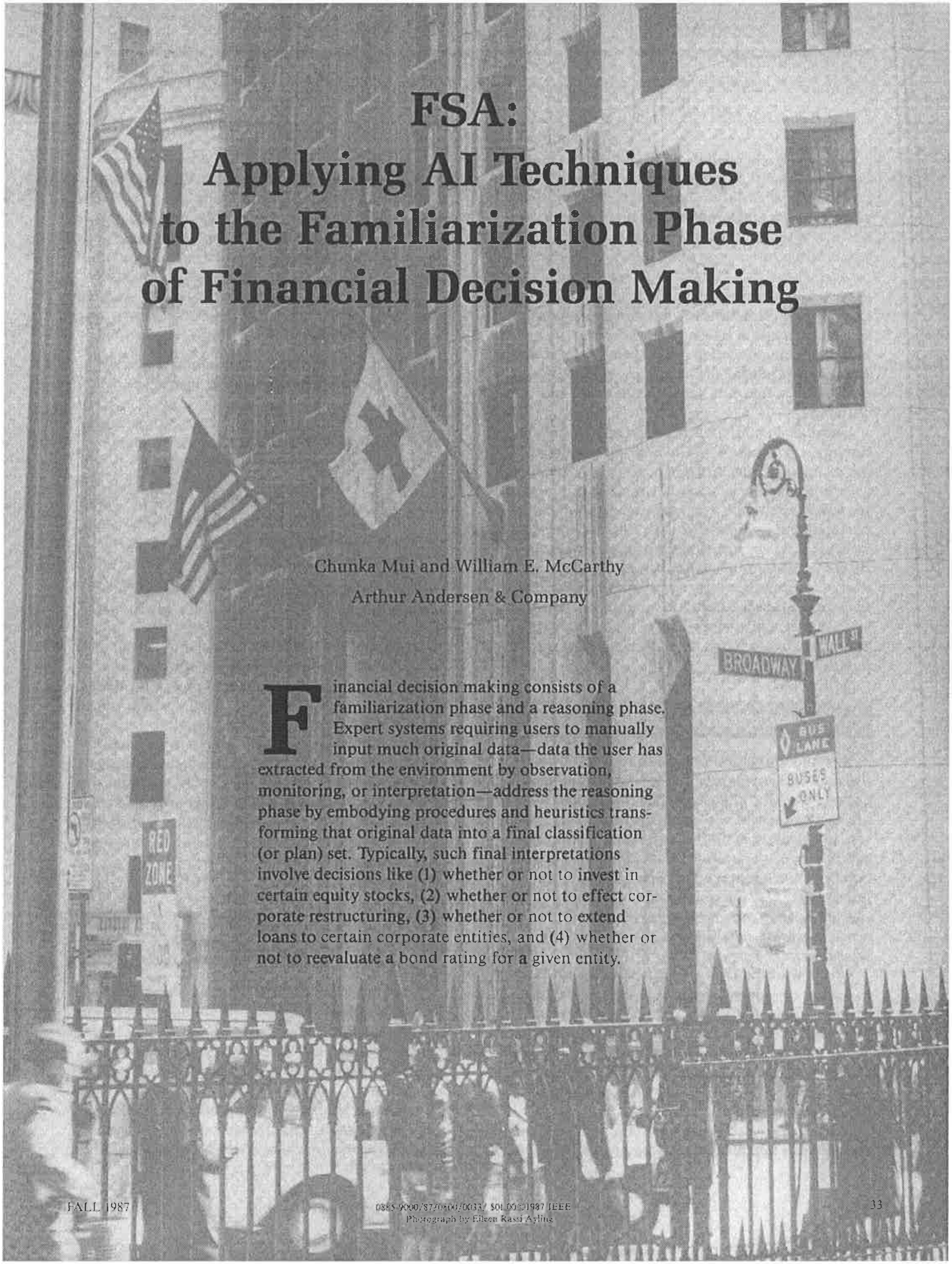
FSA: Applying AI Techniques to the Familiarization Phase of Financial Decision Making



Chunka Mui and William E. McCarthy
Arthur Andersen & Company

Financial decision making consists of a familiarization phase and a reasoning phase. Expert systems requiring users to manually input much original data—data the user has extracted from the environment by observation, monitoring, or interpretation—address the reasoning phase by embodying procedures and heuristics transforming that original data into a final classification (or plan) set. Typically, such final interpretations involve decisions like (1) whether or not to invest in certain equity stocks, (2) whether or not to effect corporate restructuring, (3) whether or not to extend loans to certain corporate entities, and (4) whether or not to reevaluate a bond rating for a given entity.

0885-9000/87/0800/0033/ $01.00 ©1987 IEEE
Photograph by Eileen Kassi Ayling

By contrast, this article will demonstrate AI techniques in support of the familiarization phase. More specifically, we will show how relatively free-form financial filings (unstructured financial statements and footnotes, for example) can be captured with natural language processing techniques and interpreted for further use with knowledge representation formalisms such as inheritance hierarchies. We will explain how we used these ideas to design and construct FSA—the financial statement analyzer—an AI system built by Arthur Andersen & Company under contract from the SEC during the pilot phase of the EDGAR project (electronic data gathering, analysis, and retrieval).

Before explaining specific details of FSA's operation, we will discuss financial decision making and the use of financial statements in general. We believe such general discussion will convince readers of the need for widely applicable front-end processing of corporate data such as balance sheets and income statements that, along with other mandated government filings, constitute an invaluable resource for financial analysis. Surveys assessing information sources for individual investors, institutional investors, and financial analysts confirm this view by consistently giving corporate annual filings the highest rankings of importance.[1]

Financial decision making

Human information processing theorists hypothesize that decision making divides into a number of phases.[2] The first phase embraces those cognitive activities concerned with explicitly recognizing relevant information in the decision maker's environment, and is referred to by terms like "information acquisition" or "intelligence." This first step can include some preliminary interpretive data processing; however, its primary purpose is to set the stage for later problem-solving activities that process first interpretations into final decisions.

This initial exploratory activity was recently studied in a series of process-tracing experiments.[3-6] Each of these experiments asked experienced subjects to talk aloud while solving a significant financial problem. Protocols transcribed from these recorded sessions showed extensive important use of financial statement data. Figure 1 illustrates Bouwman's explanation of this process.[4]

Bouwman divided financial decision-making into two phases—familiarizing and reasoning. His subjects included two groups of experts making two types of decisions. Financial analysts evaluating a stock for possible investment formed the first group, while loan officers evaluating a multimillion-dollar participation loan formed the second. As Figure 1's "clouds" illustrate, analyzing financial statements and calculating financial ratios comprised significant information usage in both phases—but in the first phase especially.

According to Bouwman, as Figure 1 portrays, financial decision making consists of the following steps:

(1) Scanning the environment and background values to identify key items such as "sales" or "net income";
(2) Evoking financial templates for companies or industries from long-term memory (for example, "high-tech company" or "late recessionary industry");
(3) Searching for instantiations of these templates with specific information (that is, a more directed reading of initial data); and
(4) Evaluating, or deciding overall.

In rough terms, the first step corresponds to familiarizing—while the second, third, and fourth accrue to reasoning. The instantiations called for explain the continued use of financial statement and ratio data throughout the entire decision process.

Bouwman developed this description of the decision process after several experimental studies of financial decision-making. We would not attempt to generalize Bouwman's two-phase description to cover most actual users; however, we do believe it constitutes a widely applicable framework for analyzing problems involved in providing automated support for financial problem solving. The next section uses that framework in discussing AI approaches to front-end processing of corporate financial data.

Standardized processing for familiarization purposes

Surveys of financial decision-makers and detailed laboratory studies of those decision-makers in action establish the primacy of financial statement use in investment, loan, and restructuring decisions. However, this primacy leads to an interesting data availability paradox.

On the one hand, the amount of presently available corporate financial data is clearly overwhelming. For instance, the required corporate filings with the SEC

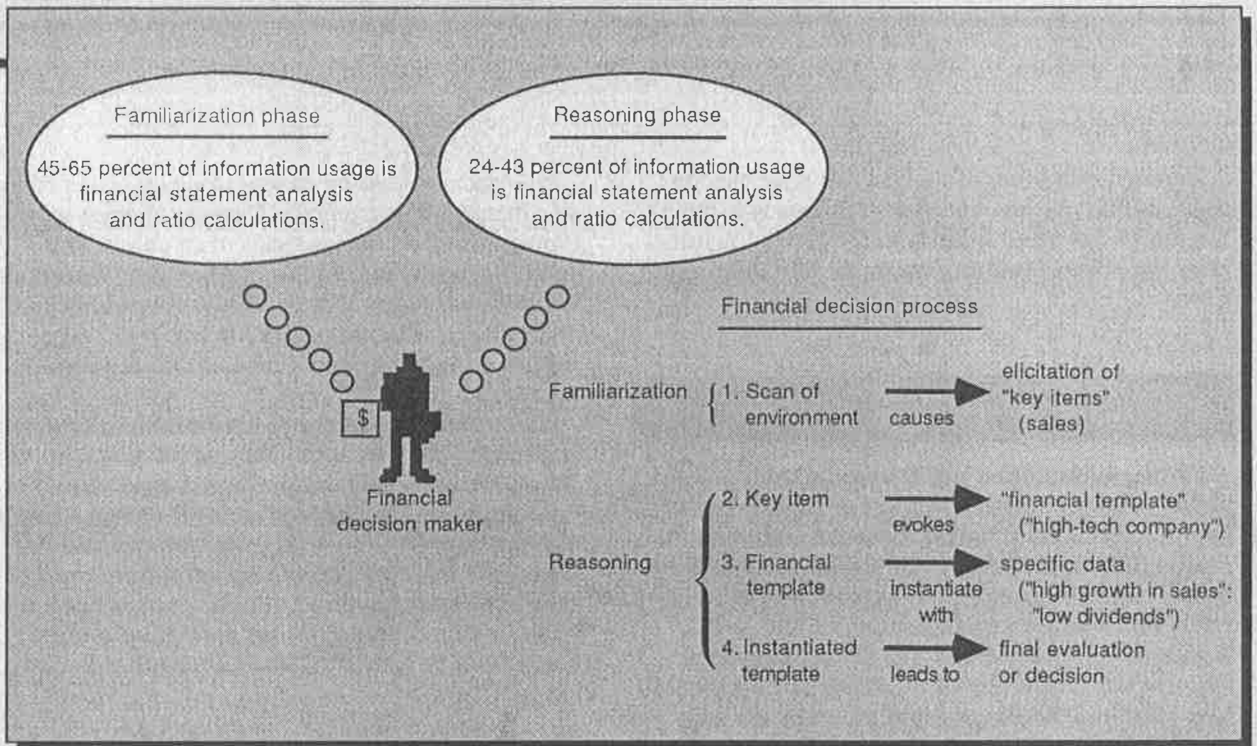


Figure 1. The financial decision process.

exceed four-million pages annually. This viewpoint is best expressed in the following analysis of a previously cited process-tracing study:

> Individuals who, as a result of their professional position, act as proxies for other investors, all felt inundated with [corporate financial] information they felt was of questionable value.[7]

On the other hand, present financial data sources are inadequate in many respects. Research has noted that this judgment applies even to the vast array of computerized financial data services currently available.[8,9] Present sources do not adequately account for a wide range of company sizes and types, and they do not make it possible to aspire to high levels of uniform classification and retrieval for heterogeneous corporations.

The solution to these seemingly contrasting problems lies in some knowledge-based processing of disaggregated and uninterpreted financial data. While present data services work well in many cases, they would do better if their input were made more uniform and interpretive.

Such front-end processing would require considerable accounting knowledge concerning the composition of financial statement numbers, and it would also require the ability to extract those accounting numbers (and related conceptual ideas) from unformatted text-like footnotes and proxy statements. Such interpretive and extractive processing is a formidable task and is difficult to build into a computerized system—which probably accounts for the fact that most present-day financial expert systems aim at decision support for problem solving's reasoning phase rather than its familiarization phase. Such reasoning support systems are not uniform because they attempt to emulate a disparate group of expert decision makers. Support for familiarization can be uniform, however. In fact, the SEC's EDGAR project sought such support.

EDGAR

The EDGAR pilot system enabled volunteer corporations to submit required filings electronically (as opposed to paper submissions). While developing this pilot, the SEC wanted to assess the ultimate feasibility of an automated system that would include some of the types of interpretive and extractive processing discussed above. Because their legislatively mandated goal is to make all filed information on registered corporations available to the trading public, the SEC was exploring the potential for a processing system that would monitor filings for compliance with securities laws and also convert that nonuniform data into easily accessible information.

During development of the EDGAR pilot, the SEC commissioned Arthur Andersen & Company to build two knowledge-based systems: ELOISE (the English language-oriented indexing system for EDGAR), and FSA.

The financial statement analyzer

FSA represents a first step at applying AI techniques to the familiarization phase of financial analysis. It performs ratio analysis using corporate annual reports (10K) as the information source. Building the system required approximately 18 man-months of effort with a project team consisting of SEC and Arthur Andersen personnel. The development environment was IntelliCorp's KEE running on Symbolics Lisp Machines. Results of this work are on file with the SEC.[10]

An object-oriented system, FSA's structure is modeled after the accounting domain's knowledge structure. Explicit knowledge representation and natural language processing techniques were molded to technical requirements imposed by the problem domain. In order to extract key financial data from corporate annual reports, the system had to systematically interpret tabular financial statements and textual footnotes. FSA explicitly represents accounting knowledge needed to understand financial statements and financial knowledge needed to perform ratio analysis. It incorporates natural language processing techniques to parse textual footnotes.

Analysis is organized using a message-passing control structure. Each financial statement item is a computational object having a local state (composed of slots) and operators (represented as methods) and communicating via message. Each object must find itself within financial documents—a responsibility invoked via message to the object's FIND-YOURSELF method. A ratio (such as QUICK-RATIO) receiving a FIND-YOURSELF message sends a FIND-YOURSELF message to each item in its formula, waits for replies, and then applies these values to the formula. A statement item (such as RENTAL-EXPENSE) receiving a FIND-YOURSELF message would search the company's financial documents for its value. In the case of RENTAL-EXPENSE, the object would search the Income Statement and, if that failed, the textual footnotes. The nature and complexity of this search is completely hidden from the FIND-YOURSELF message sender.

This message-passing control structure makes analysis a demand-driven process. Data search must be explicitly invoked via a message to an object. This control structure follows the Actor model.[11] The result is a modular system allowing for easy expansion and maintenance, with FSA's search behavior closely modeling the heuristics of human problem solvers in accounting.

FSA currently understands balance sheets, income statements, and their accompanying footnotes. It can extract necessary data and perform ratio analysis. For example, it uses (1) the *quick ratio* to measure a firm's ability to pay off short-term obligations without relying on inventory sale, (2) the *current debt to equity ratio* to measure how much a firm has been financed by short-term debt, and (3) the *times fixed charges earned ratio* to measure a firm's ability to pay fixed charges.

A system model

Figure 2 models FSA as it currently exists. The main system input is a company's financial documents, including statements and footnotes. Two knowledge bases support the system: One contains accounting and financial knowledge, and the other contains semantic structures (schemata) that drive footnote processing. Users are financial analysts interacting with the system to initiate queries and resolve ambiguities. For an overall grasp of FSA's functionality, one must understand how the two accounting knowledge bases are used in interpretive and extractive processing. Each is described below.

Account hierarchy. FSA uses a structured representation of accounting knowledge covering the composition of financial statements and the relationships between statement items. We need this knowledge to interpret loosely structured financial statements and to extract accurate account values from them. FSA models its financial knowledge after chart of account hierarchies used in the accounting profession.[12] Such structures constitute a standard method for organizing financial information, and we found that accounting problem solvers intuitively attempt to organize financial data into such hierarchies before trying to reason with it.

Charts of accounts lead naturally to taxonomic classification using a semantic network formalism. Figure 3 illustrates a small portion of FSA's accounting semantic network (the actual network is much

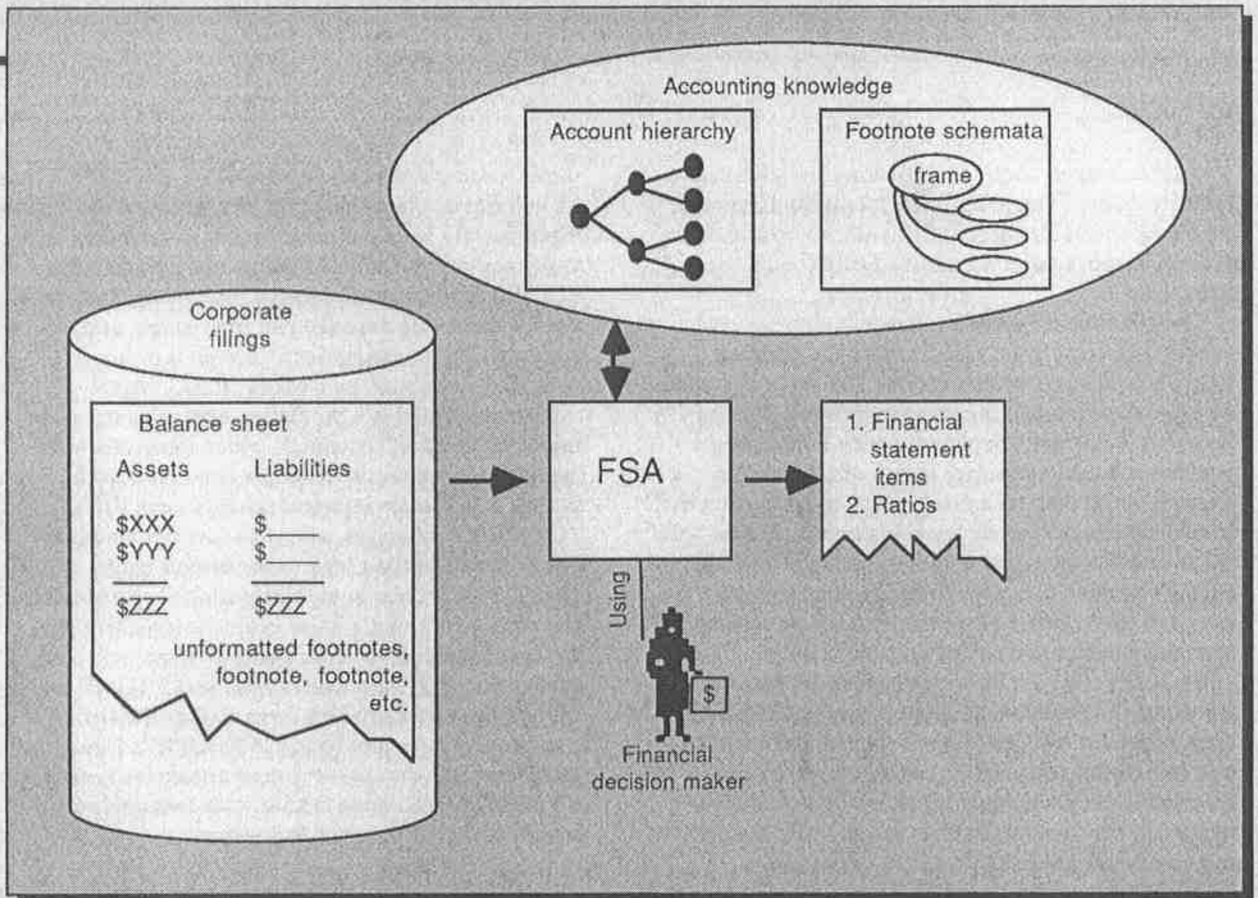


Figure 2. An overview of FSA operation.

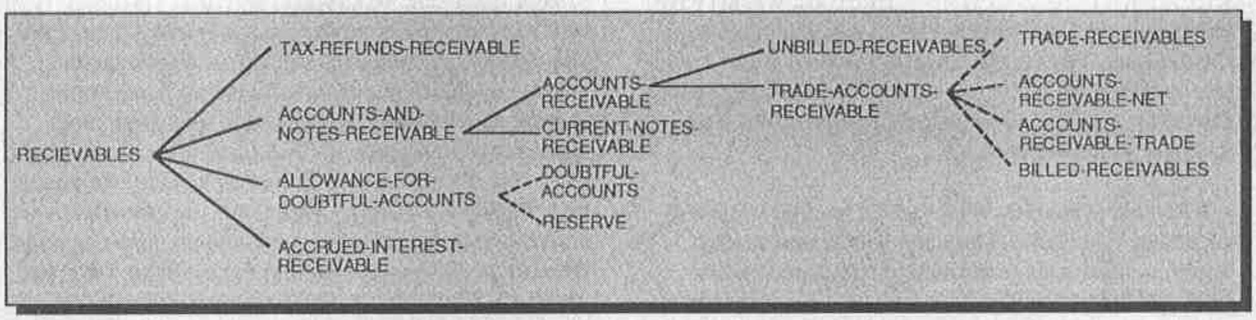


Figure 3. A subset of account hierarchy.

larger). Each node represents an object corresponding to some financial item. The taxonomic relationships within the network are SUB-ACCOUNT-OF and ISA, corresponding to the canonical SUBCLASS and INSTANCE relationships. Charts of accounts contain many levels of general accounts and subaccounts, as the SUB-ACCOUNT-OF relationship represents. The ISA relationship represents that each account may be designated in different ways. Figure 3 indicates the SUB-ACCOUNT-OF relationship with a solid arrow and the ISA relationship with a dashed arrow. For example, UNBILLED-RECEIVABLES is a subaccount of ACCOUNTS-RECEIVABLE, and RESERVE is an ALLOWANCE-FOR-DOUBTFUL-ACCOUNTS.

We reason within the semantic network via heuristic and algorithmic methods attached to each object. Methods (like FIND-YOURSELF) use the network directly to reason about aggregations and alternative interpretations. The inheritance hierarchy formed by the network enables descriptive and procedural information to move from accounts to subaccounts and instances.

The semantic network's efficacy in representing accounting knowledge was essential for eliciting knowledge from domain experts. The network could be presented in much the same way that a chart of accounts is normally depicted, which in turn maps well to an expert's intuitive image of the domain. Experts could then take direct roles in structuring the knowledge base, thereby lessening problems that arise when knowledge engineers must translate from the expert's domain language to an AI representation language. Experts could also describe their analytical methods directly in chart of account terms.

By design, this application of semantic networks exhibited the desirable qualities of representation systems proposed by Rich—representational adequacy and inferential adequacy.[13] These properties correspond to (1) adequately representing knowledge needed in the domain, and (2) successfully manipulating representational structures to derive new structures—structures replicating human inference of new knowledge from old. Our representation also displayed acquisitional efficiency—the ability to acquire new information easily.

The system's search strategy and representation structure enabled us to easily identify knowledge base deficiencies. Most omissions were new instances for the network. For version control reasons, FSA required that we insert this new information by hand. However, system-controlled acquisition could also have been implemented.

Footnote schemata. FSA faced financial statement footnotes that significantly challenged automated analysis. Footnotes tend to be unstructured collections of text and tables, with information spread over multiple sentence fragments interwoven with numerical tables. This anomalous syntactic structure foiled our attempts to build a full syntactic parser—a parser using grammar based on systematic formalisms such as augmented transition networks or charts.[14]

Consequently, in conjunction with semantic analysis, we chose to parse footnote syntax partially. We defined special objects in the semantic network, representing financial items to be found within the textual footnotes. We called these special objects *schemata,* and used them to represent accounting knowledge contained within footnotes. Figure 4 shows a sample schema for RENTAL EXPENSE that we will use in a later example. We designed these schemata directly from our experiences in designing and implementing the earlier SEC prototype ELOISE. ELOISE read corporate proxy statements, looking for certain corporate antitakeover provisions, and was patterned on the semantically driven top-down approach developed by DeJong in FRUMP.[15]

FSA uses DeJong's prediction and substantiation model to interpret footnotes, which integrates with financial statement processing's demand-driven nature. The system activates schemata via FIND-YOURSELF messages, and then invokes methods that *predict* and then try to *substantiate* each schema role. A text analyzer developed specifically to deal with a footnote's loose syntactic structure performs substantiation. FSA's text analyzer relies on weaker methods than its ELOISE and FRUMP counterparts because footnotes have neither the strong grammatical structure found in ELOISE's legal proxy statements nor the simple grammatical clarity found in FRUMP's UPI news stories. Our text analyzer proved sufficient for FSA's domain.

A typical FSA scenario

FSA provides a workbench-like user interface giving analysts control over multiple information sources and system functions. In a typical scenario, financial analysts choose a company and year to analyze. FSA can display any company document on the screen. Analysts can initiate ratio calculations at any point. During the calculation, the system may alert users when certain anomalous conditions arise. If the company had no cash assets, for example, analysts would be notified immediately. Results of the calculation are shown beside source documents within 30-60 seconds (further performance statistics are available from the authors). Hardcopy session logs are also available. Analysts control the conditions under which the system will query and the detail level of system responses ranging from alerting on all ambiguous conditions and showing all processing details to never alerting and showing only final ratio values.

Currently, the system is designed to interact with the financial decision maker. However, we believe FSA should ultimately be a background monitoring system. In this monitoring mode, it would support financial decision making's familiarization and (in part) reasoning phases. It would scan all incoming documents for relevant information, perform limited interpretive

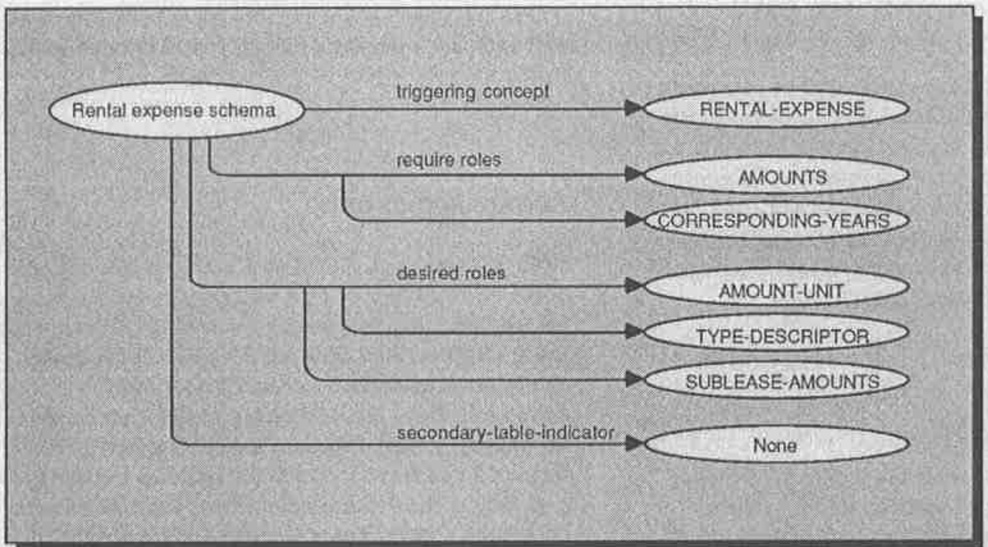


Figure 4. A rental expense schema.

processing on that information, and channel its results to a more comprehensively interpretive mechanism. Today, such interpretation is done by a human analyst; in the future, an expert system could do it.

To illustrate FSA's actual operation, we will display instantiations of two account elements—accounts receivable and rental expense. Accounts receivable is commonly instantiated through accounting hierarchy knowledge, while rental expense typically involves both hierarchy and footnote schema knowledge.

Accounts receivable. When financial decision-makers analyze a company, accounts receivable is of interest either by itself or as a component of a more aggregate figure like the quick ratio (a surrogate for corporate liquidity, as it estimates a company's ability to meet short-term financial obligations). Decision makers interested in assessing loan or equity potential can compare such surrogates across companies.

When users opt to see a QUICK-RATIO, FSA sends a message to all numerical components of that formula to find themselves in the financial statements using structures resembling Figure 3's. The search range for that particular ratio's numerator would be defined as CURRENT-ASSETS (not shown in Figure 3) and one of its key components would be ACCOUNTS-RECEIVABLE. If that particular account is not found, FSA will go up or down the knowledge hierarchy and offer either generalization to a parent or combination of subclasses as an alternative. For this particular element, analysts will often accept just TRADE-ACCOUNTS-RECEIVABLE as a substitute because it normally constitutes most of ACCOUNTS-RECEIVABLE.

Rental expense. Unlike ACCOUNTS-RECEIVABLE, instantiating items like RENTAL-EXPENSE frequently requires digging information out of financial statement footnotes. When this account figure is not found in corporate income statements, the account object issues a message to find itself with the aid of frame-oriented knowledge structures encoded in Figure 4's schema. The account name serves as the "triggering concept" for calling the schema and fills the first slot. The "required roles" slot indicates concept attributes required for instantiation (AMOUNTS and CORRESPONDING-YEARS), while the "desired roles" slot designates optional items used in rent expense calculations.

This schema might interpret the following sentence:

> Gross rental expense amounted to $36,238,000 for 1983, $33,467,000 for 1982, and $29,046,000 for 1981, which was reduced by sublease income of $3,114,000,

$3,582,000, and $2,578,000 in 1983, 1982, and 1981.

A rental expense of $29,885,000 would be returned for 1982 after this schema was invoked. We did not use Figure 4's final slot in this example, but we often need that slot when sentence fragments in footnotes indicate that information must be gleaned from tabular data. If such tabular processing is required, footnote interpretation reverts to parsing techniques used for the bodies of the financial statements.

At the outset, we made a case for directing significant AI system building effort towards the front-end or initial phases of financial problem-solving. We based our case on evidence gleaned from empirical laboratory studies and surveys of financial decision-makers. We envision that the somewhat standardized and wide-ranging interpretive processing such systems could accomplish might serve the needs of many different reasoning phase decision-support tools, including some expert systems.

The second part of this article described FSA, focusing on its goal of making familiarization phase information from financial statements more accessible. Our descriptions concentrated on the successful application of AI techniques to codifying specific accounting knowledge structures such as account hierarchies and footnote schemata.

Since the SEC's EDGAR project remains in its pilot phase, no final cost-benefit decisions have been made regarding the ultimate feasibility of fully implementing an FSA-like system. Such a decision would require candid assessment of "the scaling problem" for this particular domain.[16] Judgments would have to made as to how well techniques used and lessons learned in the pilot would work on larger search spaces involving the following extensions:

(1) An ability to process additional financial statements and to calculate additional ratios,

(2) An expanded capability to accommodate both opportunistic and background processing in addition to FSA's present abilities with demand-driven query processing, and

(3) An augmented accounting vocabulary that includes terminology from a wider variety of industries.

Our experience indicates that AI approaches work well and that a similarly designed operational system in this domain would be successful. ■

Acknowledgments

We acknowledge the help of many people involved in the development of FSA and this article. In particular, we thank Bruce Johnson for having the vision that made FSA possible; Janice Bibbs, Randy Weisman, and Susan Zeckendorf for their major roles in developing FSA; Joe Aleknavag and Robert Bayless for providing their financial analysis expertise; Rich Baader, Tilda Brown, and Mike DeBellis for their support during the development effort; John Lawler for his graphics work; Larry Downes for his insightful commentary on this article; and M.J. Bouwman and Matthew Anderson for their theoretical directions. We also thank Beth Quevillon and the entire AI group at Arthur Andersen & Company for their continuing support and motivation.

References

1. L. Chang, K.S. Most, and C.W. Brain, "The Utility of Annual Reports: An International Study," *J. Int'l Business,* Spring/Summer 1983, pp. 63-84.
2. H.A. Simon, *The New Science of Management Decision,* Harper Brothers, New York, N.Y., 1960.
3. K. Ericsson and H. Simon, "Verbal Reports as Data," *Psychological Review,*, Vol. 87, No. 3, 1980, pp. 215-251.
4. M.J. Bouwman, "Towards Expert Systems: The Analysis of Expert Financial Behavior," tech. report, Dept. of Accounting, University of Oregon, Eugene, Ore., 1985.
5. S.F. Biggs, "Financial Analysts' Information Search in the Assessment of Corporate Earning Power," *Accounting, Organizations and Society,* Vol. 9, No. 3/4, 1984, pp. 313-323.
6. M.J. Anderson, "Some Evidence on the Effect of Verbalization on Process," *J. Accounting Research,* Vol. 23, No. 2, 1985, pp. 843-852.
7. M.J. Anderson, *The Investment Decision—An Analysis Using Verbal Protocols,* PhD dissertation, Michigan State Univ., E. Lansing, Mich., 1982, p. 85.
8. R.B. McElreath and C.D. Wiggins, "Using the Compustat Tapes in Financial Research: Problems and Solutions," *Financial Analysts J.,* Jan./Feb. 1984, pp. 71-76.
9. G. Foster, *Financial Statement Analysis,* Prentice-Hall, Englewood Cliffs, N.J., 1986.
10. "Final Report on the Financial Statement Analyzer to the Securities and Exchange Commission," tech. report, Arthur Andersen & Company, Chicago, Ill., Dec. 1985.

11. C.E. Hewitt, "Viewing Control Structures as Patterns of Passing Messages," *Artificial Intelligence,* Vol. 8, No. 3, pp. 323-64.

12. B.E. Cushing, *Accounting Information Systems and Business Organizations,* Addison-Wesley, Reading, Mass., 1982.

13. E. Rich, *Artificial Intelligence,* McGraw-Hill, New York, N.Y., 1983, Ch. 7.

14. T. Winograd, *Language as a Cognitive Process: Syntax,* Vol. 1, Addison-Wesley, Reading, Mass., 1983.

15. G.F. DeJong II, *Skimming Stories in Real Time: An Experiment in Integrated Understanding,* PhD dissertation, Yale Univ., New Haven, Conn., 1979.

16. D.A. Waterman, *A Guide to Expert Systems,* Addison-Wesley, Reading, Mass., 1986, pp. 27-28.

Chunka Mui is a senior staff member in the AI Group of Arthur Andersen & Company's Technical Services Organization. He was lead designer of the financial statement analyzer and also designed and implemented parts of ELOISE. Besides financial applications of AI, his research interests include the application of AI to software engineering. Before joining Arthur Andersen & Company, he was a member of the Message Passing Semantics Group at MIT's AI laboratory. He received his BS in computer science and engineering from MIT in 1984. He is a member of the ACM, the Computer Society of the IEEE, and AAAI. His address is Arthur Andersen & Co., 33 West Monroe, Chicago, IL 60603.

William E. McCarthy is a member of Arthur Andersen & Company's AI Group, where he is working on projects dealing with automated knowledge acquisition and expert systems methodology. He is on leave from his position as professor of accounting in the graduate school of business administration at Michigan State University. His primary research interests at MSU have concerned the semantic modeling of accounting transaction systems, the computational modeling of expert behavior, and the linkages between AI and database systems. He received his AB in economics from Boston College and his PhD in accounting and computer science from the University of Massachusetts at Amherst. His permanent address is 329 Eppley Center, Michigan State University, East Lansing, MI 48824.