8

Journal of Accounting Education
Vol. 3, No. 2
Fall 1985

EDP AUDITING INSTRUCTION USING AN INTERACTIVE GENERALIZED AUDIT SOFTWARE

Miklos A. Vasarhely
COLUMBIA UNIVERSITY
and
W. Thomas Lin
UNIVERSITY SOUTHERN CALIFORNIA

Abstract: Most auditing curricula do not provide students with experiential reinforcement of auditing and/or computer auditing concepts. This paper reports on an interactive generalized audit software package that has been designed to fill this educational void. The TREAT package consists of four different modes of operation with four audit cases that enable students to gain hands-on audit experience. The features of the TREAT package are similar to many accounting firms generalized audit software packages. Students using the package and its audit cases have proven TREAT to be an effective and efficient pedagogical tool.

INTRODUCTION

Auditing has been defined as a systematic process of objectively obtaining and evaluating evidence regarding assertions about economic actions and events to ascertain the degree of correspondence between these assertions and established criteria and communicating the results to interested users [AAA, 1973, p. 2].

The primary criteria or objectives of auditing include asset safeguarding, data integrity, system efficiency and effectiveness [Weber, 1982 pp. 7-9]. Use of a computer for data processing does not affect the basic auditing objectives; however it affects how these objectives must be achieved. For example, the AICPA's Statement on Auditing Standards No. 48, "The Effects of Computer Processing on the Examination of Financial Statements," states that the auditor should consider the methods the entity uses to process accounting information in planning the audit because such methods influence the design of the accounting system and the nature of the internal accounting control procedures [AICPAS 1984, p. 4]. It recommends the auditor to consider the use of computer-assisted audit techniques to increase the efficiency of performing audit procedures. In addition, the auditor should consider whether specialized skills are needed to consider the effect of computer processing on the audit, to understand the flow of transactions, or to design and perform audit procedures. If specialized skills are needed, the auditor should seek the advice of a professional possessing such skills. [AICPA, (1984, p. 3)].

In today's auditing environment, the computer has been used substantially in the following two areas: (1) to test the information processing system internal control procedures, i.e., compliance testing, and (2) test the details of transactions and balances, i.e., substantive testing. General audit software (GAS) can be used by the auditor to conduct compliance testing (sample selection and analysis), as well as substantive testing (footings and extensions, audit data comparison, and confirmations).

put comma in
pull parentheses out

GAS has been used most often for substantive testing. Since GAS is the most often and the most repeated use of the computer as an audit tool in EDP audit practices [Perry and Adams (1978], this article will concentrate on the use of GAS in EDP audit education.

A GAS is a computer program, or a series of computer programs, specifically designed to perform certain audit-related data processing functions. These functions include reading computer files, selecting desired information, performing calculations, and printing reports in a format specified by the auditor [AICPA 1979, p. 12]. Many accounting instructors have incorporated such GAS packages into their audit curricula. Sardinas (1979) presented a collection of EDP audit course contents from various universities that illustrated this trend.

The use of GAS packages in an educational environment requires an experiential learning philosophy. Students actually simulate parts of the audit process by examining computer generated records. This philosophy is applied by many instructors who simulate traditional audit situations by using comprehensive audit cases.

This article discusses the content and use of an interactive GAS package for auditing education. The instruction of EDP audit is initially examined and the experiences are described that led to the development and utilization of the interactive GAS package. The last sections will concentrate on the package's system features, use, and educational considerations.

COMPUTER AUDIT AND ITS INSTRUCTION

There has been an exponential increase in the use of computers within business organizations. The proliferation of computer uses in business environments has introduced new concerns for auditors and accountants. Records, once accessible, traceable and difficult to adulterate, have become inaccessible, difficult to trace and subject to tampering. Computer auditing has evolved in response to these trends, as auditors have capitalized on the economics of the use of data processing as a labor saving device.

The field of computer auditing achieved notoriety after the unfortunate events that led to the Equity Funding scandal [Seidler, Andrews and Epstein, 1977]. These events and many other computer fraud cases as well as the proliferation of computer uses by the business and public sector led to increasing demands for skilled computer audit personnel.

The major source of personnel for public accounting firms is the recent graduates of accounting programs in major universities. Computer audit personnel may be classified into two main groups: (1) computer personnel trained in the audit functions, and (2) audit personnel with supplemental training in data processing. Some business schools are preparing graduates versed in both fields.

In recent years more and more universities have reacted to the business and accounting environments by offering either an advanced auditing or an EDP auditing course. The instructor usually selects one of the six EDP auditing textbooks (Burch and Sardinas, 1978; Davis et al., 1983; Mair et al., 1978; Porter and Perry, 1984; Watne and Turney, 1984; Weber, 1982) and supplements with a GAS to let students have hands-on experience. The typical course content would include:

1. The audit and the computer
2. EDP concepts and flowcharting techniques
3. A general approach to EDP auditing
4. General controls in an EDP environment
5. Computer application controls
6. Computer audit tools and techniques: GAS
7. Computer audit tools and techniques: test data techniques utility programs, code review
8. Concurrent auditing techniques
9. Control and audit of mini- and micro-computer systems.

The teaching methods include lectures, case discussions, student hands-on experience with a GAS, guest speakers, and a term project. There are four major steps in an EDP audit:

1. Understand the significant accounting applications including the flow of transactions and accounting controls.
2. Review of the computer installation including both general controls and application controls. General controls apply to all computerized activities within the information system. They include organization and operations controls, hardware and controls, hardware and software controls, access controls, data and physical security and back up, systems development and maintenance controls. Application controls serve to achieve control objectives relating to the special characteristics of business and accounting systems such as revenue, purchasing, production and cash transaction systems. They include data capture and input preparation controls, input controls, processing controls, and output controls.
3. Test compliance to determine whether or not the system of internal controls operates as it is purported to operate.
4. Test the details of the records and conduct analytical review procedures to obtain sufficient evidence so the auditor can make a final judgment on whether or not material losses have occured or could occur during computer data processing.

GAS is used most often for testing the details of the records and conducting analytical review procedures. It can be used for substantive testing of transactions and account balances such as cash receipts and disbursements, payroll, sales, accounts receivable, inventory, fixed assets, payable. For example, in a payroll system the auditor can use a GAS to: (1) print and foot payroll transactions, (2) summarize payroll transactions by the respective account distribution for reconciliation to the general ledger and inventory charges, (3) test computation extensions and deductions, (4) merge the payroll transaction files with the payroll master files, and test for exceptions such as unusual number of exemptions, pay code, pay rate, gross pay, and hours worked.

A CASE HISTORY OF THE TREAT SYSTEM

In 1971 the University of Southern California added a computer auditing course to the accounting curriculum, and subsequently experimented with different approaches to its instruction. This experimentation has involved using two major GAS packages, resulting in the development of the TREAT (Terminal Related Educational Audit Tool) system.

Touche Ross' STRATA package was initially used in the USC curriculum for seven years by three different instructors. The Deloitte, Haskins and Sells AUDITAPE package was used on a more limited basis. This permitted experimentation with many alternative teaching approaches from extensive lectures on software package features to complete application audits. It led to the identification of a key set of problems:

1. Batch system turnaround time proved cumbersome in an educational context.
2. Students became confused with extensive card manipulation.
3. Long waits for batch jobs caused considerable interruptic n in the student's learning process.
4. Audit software with too many features led to confusion on the students' part.
5. A great majority of the problems were with syntax and not with the package, as students spent far too much time learning syntax idiosyncrasies.

In addition to these problems, a set of additional educational objectives were identified:

—Students needed greater understanding of the nature and problems concerning computerized business files.

—Students should learn how to document their findings.

—Students needed to experiment with different solutions of audit problems.

—Students should learn the specific problems related to different applications (e.g., Payroll, Accounts Payable, Inventories, etc.)

These problems and the resulting objectives led to the development of a system with the characteristics described in the following section. The perception of these problems was not unique to USC. Professors at other universities had also developed instructional audit softwares of a similar nature. (E.g., Professor James Lampe's MARS system at Missouri and Professor Hart Will's ACL at the University of British Columbia). A brief description of these systems can be seen in the technical report of the AAA's Audit Section (AAA, 1983).

INTERACTIVE SYSTEM FEATURES

The GAS package must serve as an adequate introduction to the computer audit software that students will find on the job. Most computer audit software packages have reasonably similar features (Morton 1982; Perry 1981). In developing TREAT, the specific characteristics of the following three systems were examined: Deloitte, Haskins and Sells' AUDITAPE, Arthur Anderson's AUDEX and Touche Ross's STRATA.

TREAT is STRATA based (emulating STRATA's main features), simplified in nature, language independent from the user standpoint, interactive, and a user of small simplified client files (databases). The source code is fully compatible with STRATA allowing programs to be tested in an online mode and also executed later under a batch version of STRATA. Thus, after introductory GAS instruction, students can proceed to more advanced use of the STRATA package.

The TREAT system is composed of four micromodules (modes of operation):

1. DEFINITION MODE—to define STRATA type functions and edit these at entry.
2. EXECUTION MODE—to execute TREAT programs and al'ow suspended execution in case of non-fatal error.
3. LIST MODE—to provide an annotated listing of TREAT programs for debugging and documentation purposes (similar to STRATA's diagnostic).
4. MODIFICATION MODE—to redefine pages or lines of TREAT code.

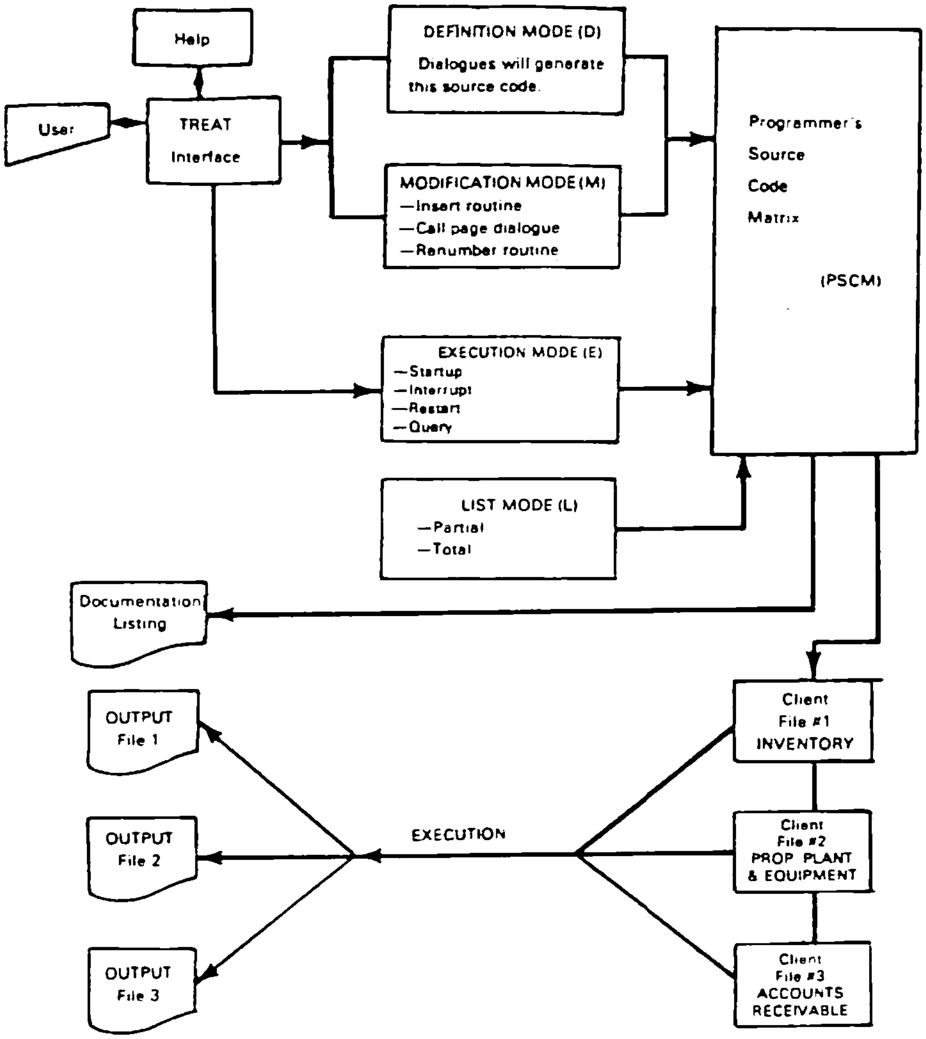
The key functions of the system (as well as of other GAS's) encompasses: encompass

- Auditor file definition—to extract and format client's records for the use of the auditor.
- Mathematical functions—to allow mathematical manipulation of variables.
- Sampling functions—to allow extraction of samples from client files obeying different sampling plans.
- Sorting function—to sort files by designated fields.
- Summarization function—to aggregate records with common characteristics and perform footing operations.

—Report Generation function—to prepare standardized reports, confirmations, scales, etc.

Students, once instructed on these functions, flowchart and audit application and a TREAT program. The TREAT program is then defined through a terminal, which generates the Programmer's Source Code Matrix (PSCM in Figure 1) containing computer understandable instruction. Definition involves the manual utilization of available functions to perform the desired audit tasks. This is followed by EXECUTION which detects errors and, finally, desired changes are done in the MODIFICATION phase of the program. At certain points students use the LIST mode to get an up-to-date listing of their TREAT program.

Figure 1: Symbolic System Representation



The software is APL based allowing relatively easy adaptation to different computer installations. The only requirement for TREAT installation is the availability of a time-shared mainframe[1] with an APL interpreter. APL character set terminals are not necessary for TREAT utilization but it may be desirable to have one accessible for system installation and faculty utilization. The TREAT system is currently installed in about thirty universities in the United States and abroad, and is operational on a large number of different computers (e.g. IBM 4341, VAX 11/780, DEC 20). The system is available free of charge to academic insitutions of not-for-profit nature.[2]

A typical student would follow the ensuing steps in the design and implementation of an application:

a. Design a general application flowchart and detailed logic flowchart.
b. Completion of paged coding sheets.
c. Definition in interactive mode.
d. Execution
e. Interactive program modification
f. Execution of completed application
g. Utilization of LIST MODE for report documentation
h. Option to dump applications onto cards for execution via a batch mode.

Faculty members using the system may save substantial computer resources by storing the basic system under an instructor account and having the students (or groups of students) only store their own applications and a small "call-the-system" program. All data files are provided ready to be accessed by the students and sample TREAT programs can be used by the instructor as problem solutions [Vasarhelyi and Lin, 1979].[3] The use of a GAS in different pedagogical settings is discussed in the next section of this paper.

CLASSROOM USAGE OF THE GAS

The main educational objectives of using generalized audit softwares in the classroom are:

1. to teach and illustrate the use of the computer in an audit;
2. to allow the student to simulate the real world audit process.

The first objective raises the problems of software specificity. Instruction should not be limited to a particular software and each software package presented should serve to exemplify general features of a family of softwares.

[1] Two microcomputer based versions of TREAT are currently experimental. One, called PC-GAS, captures the essential features of TREAT using Ashton-Tate's DBASE II and a few interface programs. The second is on IBM PC based version of TREAT requiring the PC's APL implementation.

[2] Requests for software and further information should be directed to Miklos A. Vasarhelyi, 612 Uris Hall, Columbia University, New York, N.Y. 10027 specifying the type of computer to be used and the way that the tape should be written.

[3] Instructors do not need to be computer experts but must have initial help to set up the system. About 20 hours of instructor time is necessary for learning, if he/she has no prior TREAT or STRATA experience

Thus, instructors must be able to compare software features and discuss tradeoffs. Some of these comparisons are discussed in the audit section technical report [AAA, 1983]. Cases and simulated computer files can help in feature analysis and comparative discussions.

The second learning objective requires that the files used by the students be a good representation of the ones they will find in real life. In addition, the environment must be such that students can experiment with their programs, find planted discrepancies and learn to search for and evaluate evidence. In addition, this approach is a good way to teach proper working paper techniques. The need for clear documentation becomes apparent to the student. Review, evaluation and project management can be included in the learning process.

The instruction of audit techniques has expanded substantially in most US universities. Computer auditing is often required for accounting majors. The tendency for a second course (e.g. CUNY's Baruch College) and, in a few cases, even a third course in audit is becoming prevalent. Advanced auditing courses tend to concentrate on computer audit, statistical sampling, risk assessment and internal control evaluation techniques.

The scope and intensity of usage of GAS must relate to the structure of the accounting/audit curricula. Softwares used in a first audit course, must be restricted in scope due to time limitations. For this purpose TREAT includes an inventory audit case that can be used as a walk-through. This allows coverage of GAS in about one to two weeks depending on the level of detail desired by the instructor. A walk-through entails the description of the problem, the preparation of problem flowcharts, a brief description of the system's features, preparation of coded sheets, hands-on experience by the students and finally, debriefing. The walk-through is typically followed by in depth case studies using a scenario and files provided by the instructor.

TREAT encompasses three cases. The inventory case requires the calculation of extended cost and the preparation of reports for taking physical inventory as well as an exception report for unit cost. The property plant and equipment case is slightly more computational employing depreciation and data exception. Finally, the accounts receivable case is even more complex requiring aging of receivables, summarization of transactions and application of payments. An extension of the accounts receivables case illustrates the updating of Master by Transaction files.

A more detailed coverage of the software in a second or third course would initially involve two weeks of classroom learning, using the inventory walk-through described above. This would be followed by two to four weeks of assignments to explore such files as property plant and equipment, and accounts receivable data. As these assignments are performed by the students, class meetings are typically dedicated to debriefing of assignments and other topics.

Figure 2 lists specific topics covered in the steps of the process of usage of GAS in audit instruction.

Figure 2: GAS USAGE AND LEARNING

STEP	TOPIC
1 problem description and specification	• nature of audit problem • documentation of DP files • general application flowcharts • additional investigation questions
2 problem solution flowcharts	• symbolic problem solution descriptions • programming logic flowcharts • audit programming logic
3 language description	• GAS language features • specific language features • power of data processing
4 preparation of coding sheets	• specific syntaxes • nature of DP files • rigidities and limitations of DP languages
5 hands on terminal interaction	• debugging techniques • logical "honesty" • time estimates • ad hoc problem exploration
6 debriefing	• nature of the total experience • what to expect in "real life"

These two approaches (using TREAT either in the first or second audit courses) have been tested by different instructors at several institutions. The following section gives a synopsis of comments and experiences tabulated over five years of TREAT usage.

EVALUATION

TREAT is cheaper and less cumbersome for students to use than most public accounting GAS's (e.g. STRATA and AUDITAPE). It lacks however, some of the finer features of STRATA and the reality of dealing with large files of sequential data. In addition, public accounting firms will typically provide some help to the academics using their GAS's. In several instances, however, Touche Ross & Co. has helped the academics in the usage of TREAT.

TREAT's interactive mode substantially improves the quality of the learning experience by eliminating cumbersome card manipulation, decreasing the frequency of syntax errors, decreasing turnaround time and allowing rapid development, execution, and modification. This permits students to decrease emphasis on software specific technicalities and to increase emphasis on the audit problem.

The student's fear of the computer, the lack of availability of terminals, system idiosyncrasies partially offset some of the advantages discussed above. Students like the built-in error detection and correction facility that operates during the entry stage. They also think that TREAT is relatively easy to understand and does not require extensive knowledge of computer programming languages. In addition, the ability to rapidly cycle between program preparation, execution, documentation and modification is of great pedagogic value. The major limitations, however, of GAS use are: (1) faculty needs to invest considerable time in learning about the software, (2) the University needs an APL compiler, (3) the school has to provide sufficient terminals and (4) some disk space must be made available for the system and student program storage. This first difficulty is probably the major deterrent to the usage of computers for educational purposes both in a GAS and other frameworks.

SUMMARY

This paper discusses the integration of generalized audit softwares into the audit curriculum. More specifically, the key features of an interactive software (TREAT) and its integration into the audit curriculum were discussed as an illustration. A series of issues related to instructional technology and software design are raised. These include: the nature of GAS's, their use in audit instruction, their strengths and weaknesses, etc. The main conclusion is that the use of the interactive mode further enhances the quality of the learning experiences that come from the introduction of a GAS into audit courses.

REFERENCES

AAA, "A Statement of Basic Auditing Concepts," *AAA Committee on Basic Auditing Concepts.* (Sarasota, Florida, 1973).

AAA, "The Usage of Computers in Auditing Teaching and Research," *Report of the Technical Committee of the Audit Section.* (Sarasota, Florida, 1983).

AICPA, *Computer Assisted Audit Techniques.* (New York, 1979).

AICPA, "The Effects of Computer Processing on the Examination of Financial Statements," *Proposed Statement on Auditing Standards,* Statement on Auditing Standards, No. 48, (New York, July 1984).

Burch, J. G., Jr. and J. L. Sardinas, Jr., *Computer Control and Audit: A Total Systems Approach,* (John Wiley & Sons, New York, 1983).

Cash, J. I., Bailey, A. D. and Whinston, A. B., "A Survey of Techniques for Auditing EDP-Based Accounting Information Systems," *The Accounting Review*, (October 1977).

Davis, G. B., D. L. Adams and C. A. Schaller, *Auditing & EDP,* Second ed., *American Institute of Certified Public Accountants,* (New York, 1983).

Mair, W., D. Wood and K. Davis, *Computer Control and Audit*, Institute of Internal Auditors, (Altamonte Springs, Florida, 1978).

Morton, C., "Checking Up on Audit Software," *ICP INTERFACE Administrative & Accounting*, (Winter 1982): 27-31).

Perry, W. E., "Software Analysis for Internal Auditing," *Advanced Technology Newsletter*, (Institute of Internal Auditors, 1981): 12-16.

Perry, W. E., and D. L. Adams, "Use of Computer Audit Practices," *EDPACS*, Vol. VI, No. 5 (November 1978): 12-16.

Porter, W. T. and W. E. Perry, *EDP Controls and Auditing*, Fourth Ed., (Kent Publishing Company, Boston, Massachusetts, 1984).

Sardinas, J., *EDP Audit Curricula*, (University of Massachusetts, Amherst, 1979).

Seidler, L. J., Andrews, F. and Epstein, M., *The Equity Funding Papers*, (Wiley, 1977).

Vasarhelyi, M. A. and W. T. Lin, *TREAT: Terminal Related Educational Audit Tool*, (Touche Ross Foundation, New York, 1979).

Watne, D. A. and P. B. B. Turney, *Auditing EDP Systems*, Prentice-Hall, Inc., Englewood Cliffs, N.J., (1984).

Weber, R., *EDP Auditing: Conceptual Foundations and Practices*, (McGraw-Hill, Inc., New York, 1982).