NOTES ON THE STATE OF DATA BASE AUDIT

by

Jon D. Clark

Frederick G. Davis

Department of Accounting and Information Systems
College of Business Administration
North Texas State University
Denton, Texas 76203

INTRODUCTION

Practicing auditors largely ignored computer systems until third generation technology became common. By that time, however, the increase in sophistication and complexity of both hardware and software components was so great that catching-up was almost out-of-the-question. Since that time the rate of change of technological developments has been even greater with the advent of on-line processing, data base management systems, data base machines, distributed processing and distributed data base.

This doesn't imply, however, that auditors have been idle. The accounting profession has been active in developing audit techniques. However, when the burden of keeping abreast of data processing and information systems developments is added, the auditors have not kept up.

Auditors approach the problem of gaining reasonable assurance of data integrity by realizing that they can audit the system processing or audit the results of processing (the data). The first method involves an evaluation of controls and testing of these controls (called compliance testing).

The second method involves sampling of the data and testing its validity (called substantive testing). The auditor's problem when the data resides

in a data base is how to extract the data at one point in time in order to carry out a test.

The two methods are interdependent. The more the auditor can test the controls on the processing of data, the more reliance can be placed on the system and the less testing of data is required. The complexity of data base systems introduces the problems of understanding the controls and devising methods of testing a dynamic system. Further, the methods of testing are likely to require techniques to be designed into the system at the development stage.

The purpose of this paper is to introduce audit problems to the computer science community, catalog techniques suggested and solicit suggestions for further research.

A SURVEY OF DATA BASE AUDIT TECHNIQUES

The audit techniques to be surveyed will be broken into three general classes: audit around the computer, static auditing, and dynamic auditing. Auditing around the computer essentially ignores the existence of the computer and simply tests manual controls or monitors inputs and outputs for correctness. Some of the techniques of auditing around the computer will exist in a data base environment because the audit is concerned with all controls, those outside the computer as well as those incorporated in procedures within computer processing. Static auditing involves the controls within computer processing but the techniques allow testing only at one point in time. These techniques served the auditor adequately in testing batch update processing. Finally, there exist dynamic audit techniques whose operation allows the auditor to test controls during actual processing and over time. These techniques require substantial

knowledge of procedures used, usually must be specified at the design stage and are costly to develop.

Table 1 contains most audit techniques classified into these three catagories along with a brief explanation of each and several references to the literature. Each class of techniques will be discussed in the following paragraphs.

Auditing around the computer, as far as the data base administrator (DBA) is concerned, is of little importance, since it is difficult to support an activity that largely ignores an interface with the data base. Therefore, the accounting procedures used for this class will not be discussed, although references are provided for readers who wish additional information.

Static audit techniques are of interest here. These must be used off-line or with nonproduction files. Since this class of techniques ignores program-internal procedures, there is no way for the same procedures to recognize transactions or data transfers solely for the purpose of auditing (artificial transactions). Particularly in the case of controlled processing or reprocessing there is a danger of updating files twice; in a production environment this is obviously intolerable. Program code checking and flowchart verification are very costly operationally, but should be done as part of any systems implementation effort, as well as during maintenance and/or modification of systems. Auditor use of log tapes is just beginning to be explored. The magnitude of the data available and the difficulty of selecting and extracting the desired data are posing complex problems. Perhaps the safest technique of this class is the use of generalized audit software, which was very popular several years ago.

Table 1: Data Base Audit Techniques

	Techniques	Characteristics	References
I.	Audit around the computer	Both popular and traditional methods of ignoring the computer; input typically monitored and output checked, but no attempt to understand procedures.	9,12,13,18,21 24,28,29,30,31 32,35,36,39,40 41,42,44,45,50
II.	Static audit techniques	This class of techniques also does not attempt to understand computer-internal procedures but does require that the logic of processing be understood and the computer is used to assist the auditor.	9,10,11,12,13,15 16,18,19,21,24 25,26,27,28,30 31,32,35,36,39 40,41,44,45,48 49,50,51,52,56
	Test deck	A more general and appropriate term might be "test data." This technique involves submitting artificial data, transactions, etc. and monitoring output for expected results.	1,23,34
	Concurrent processing	Detects known exception conditions and checks for output caused by them.	46
	Parallel simulation	Actual data reprocessed using auditor programs, results compared to output from normal processing. Generalized audit software may be used to prepare auditor programs.	38
	Controlled processing or reprocessing	Used to check for consistency of processing by submitting identical transactions a number of times or those that have already been processed.	4
	Generalized audit software	Used primarily for testing of data files (substantive testing); not for testing of processing. Involves a set of subroutines to extract and print data values at one point in time. Most software is not designed to interact with a DBMS.	2,5,8,14,33 37,43,45,47 49,55

Table 1: Continued

	Techniques	Characteristics	References
	Program code checking	Examination in detail of all processing logic, requiring a great deal of expertise both in the computer language and the application.	3
	Flowchart verification	Similar to program code checking, but done at the flowchart level and, thus, is less time consuming.	19
	Custom-designed computer programs	Audit software written for a specific computer/application combination. Although efficient once written, may impede maintenance and/or modification of applications.	6,46
III.	Dynamic audit techniques	This class of techniques requires some degree of understanding of the program-internal procedures and therefore tends to be much more complex and less general-purpose than the other techniques covered.	7,9,10,11,12,13 15,17,18,19,21 24,27,28,29,30 31,32,36,40,41 44,45,50,51,53 55,56
	Integrated test facility	Operates using existing procedures but on artificial files and inputs or those physically segregated from normal operations, thus referred to as "minicompany" or "dummy company" procedure.	11,22,57
	Tagging and tracing	Typically involves marked real data whose progress can be traced through the programmed procedures. May require changes in processing logic.	11,57
	Mapping	Monitors use of program-internal logic for excessive and/or rare use, either of which may indicate unfriendly or incorrect prodecures.	51
	Audit hooks	Modules written into programs which allow the auditor to extract transactions during actual processing and write them to an audit file for subsequent testing.	9,50

Table 1: Continued

Techniques	Characteristics	References
Use of log tapes	Retention of transaction logs created by the system for rollback and recovery. Auditor extracts selected information for evaluation and testing.	20,54

Actually this is an audit aid whose sampling and retrieval capabilities are controlled by the auditor -- in a sense, a decision-support system. The difficulty that this technique causes is the interface problem, because of the existence of such a wide variety of system software (including operating systems and DBMSs).

Dynamic auditing requires a rather extensive understanding, although not a complete understanding, of the program-internal procedures; therefore, this class of techniques tends to be costly to either design and/or execute. The integrated test facility and audit hooks require a great deal of system design forethought, but if invoked during regular production time may be quite reasonable from an execution-cost point of view. Tagging and tracing requires forethought during system design and, additionally, may be costly to execute, particularly if artificial transactions must be backed out of the system. Mapping is a very reasonable cost technique that may be executed during low system-use hours, although it may not tell the auditor as much about the systems operation.

DATA BASE AUDIT POTENTIALS

The data base audit picture is obscure, to say the least; many second- and third-generation techniques being applied to advanced systems, particularly data base systems, are achieving inadequate results. There is some hope, however. Three proposals are made here, each of which must be evaluated prior to application in specific cases.

One possibility, at least with regard to program validation, is the use of certified software. Primarily a concept for commercial software, whether system software or application packages, the certification might be given by an independent body capable of extensive software testing.

Certification would, to a considerable extent, relieve the burden of EDP auditors in the area of software validity. Obviously, custom systems would be nearly impossible, or prohibitive in cost, to certify if they were submitted for independent testing.

A second possibility is a standardized "audit window," a window through which the data base could be queried or sampled. Insofar as we have both standardized (principally the DBTG's DBMS) and special purpose data base systems, this concept would only likely be applied to the DBTG standardized system. By defining a specific subschema for all data to be audited, generalized packages of extensive capability but reasonable cost could be provided to DBMS users.

A third possibility is the use of techniques derived from artificial intelligence for audit purposes; there is a large body of theory available that needs only to be applied to new problems. Pattern recognition, for example, could be applied to many currently collected performance parameters to detect unusual events that indicate either the possibility of error or fraud.

To be sure, there are many other evolving technologies having audit potential in the data base area (for example, data base machine and fuzzy set theory). Here we have suggested just a few.

SUMMARY OF DATA BASE AUDIT

This paper classified data base audit techniques. In addition to discussing many techniques, a few relatively unexplored "audit potentials" were offered. An important point to be made here is that audit methodologies must be identified and considered prior to data base design and implementation, to avoid redesign and modification costs or the possibility of

incurring an audit premium resulting from an intractible data base system. The DBA is best advised to consult directly with the firm's internal and external auditors for developing audit specifications for all data base systems. The audit problem is not too different from the problems of system testing and on-going maintenance testing. Hopefully the computer science community can contribute to the resolution of the auditors needs by suggesting techniques and additional areas of potential research.

REFERENCES

1. D. L. Adams, "A Survey of Test Data Generators," EDPACS (April 1973), pp 5-9.

2. _____ "Alternatives to Computer Audit Software," Journal of Accountancy (November 1975), pp 54-57.

3. _____ "Audit Review of Program Code-1," EDPACS (August 1975), pp 1-5.

4. _____ "Audit Software--DYL--250 and DYL--260," EDPACS (July 1974), pp 7-11.

5. _____ "Audit Software Requirements," EDPACS (November 1973), pp 14-16.

6. _____ "Audit Uses of SMF Reporting and Analysis Software," EDPACS (April 1977), pp 11, 16.

7. _____ "System & Audit Aspects of the Data Dictionary," EDPACS (May 1976), pp 1-14.

8. Donald L. Adams and John F. Mullarkey, "A Survey of Audit Software," Journal of Accountancy (September 1972), pp 39-66.

9. AICPA, Management Control and Audit of Advanced EDP Systems (New York, 1977).

10. AICPA, The Auditor's Study and Evaluation of Internal Control in EDP Systems (New York, 1977).

11. AICPA Computer Auditing Subcommittee's Auditing Advanced EDP Systems Task Force, "Advanced EDP Systems and the Auditor's Concern," Journal of Accountancy (January 1975), pp 66-73.

12. Brandt Allen, "The Biggest Computer Frauds: Lessons for CPAs," Journal of Accountancy (May 1977), pp 52-62.

13. Ina D. Banks, "Internal Control of On-Line and Real-Time Computer Systems," Management Accounting (June 1977), pp 28-30, 40.

14. L. A. Bjork, Jr., "Generalized Audit Trail Requirements and Concepts for Data Base Applications," IBM Systems Journal (1975), 3, 4, pp 229-245.

15. David C. Burns and James K. Loebbeck, "Internal Control Evaluation: How the Computer Can Help," Journal of Accountancy (August 1975), pp 60-70.

16. R. G. Canning, "The 'Data Administrator' Function," EDP Analyzer (November 1972).

17. _____ "The Data Dictionary/Directory Function" EDP Analyzer (November 1974).

18. _____ "The Importance of EDP Audit and Control," EDP Analyzer (June 1977).

19. _____ "The Internal Auditor and the Computer," EDP Analyzer (March 1975).

20. _____ "Recovery in Data Base Systems," EDP Analyzer (November 1976).

21. James I. Cash, Jr., Andrew D. Bailey, Jr., and Andrew B. Whinston, "A Survey of Techniques for Auditing EDP-Based Accounting Information Systems," Accounting Review (October 1977), pp 813-842.

22. Barry R. Chaiken and William E. Perry, "ITF -- A Promising Computer Audit Technique," Journal of Accountancy (February 1973), pp 74-78.

23. A. D. Chambers, "Audit Test Packs and Computer Audit Programs," The Computer Journal (1975) 18, 2, pp 98-101.

24. James R. Davis, "EDP Control Means Total Control," Management Accounting (January 1977), pp 41-44, 47.

25. Leonard H. Fine, "Data Base: The Role of the Auditor," The South African Chartered Accountant (December 1977), pp 409-412.

26. G.A.O., Auditing Computer-Based Systems, U.S. General Accounting Office, 1979, 15 pages.

27. Bruce M. Greenwald and Gary Oberlander, "IRS Audits of EDP Systems," Management Accounting (April 1975), pp 13-15.

28. John M. Horne, "EDP Controls to Check Fraud," Management Accounting (October 1974), pp 43-46.

29. Geoff Horwitz, "Needed: A Computer Audit Philosophy," Journal of Accountancy (April 1976), pp 69-72.

30. E. G. Jancurra, Audit & Control of Computer Systems (New York: Petrocelli 1974).

31. E. G. Jancurra, ed., Computers: Auditing and Control (New York: Petrocelli Charter, 1977).

32. E. G. Jancurra, "Technical Proficiency for Auditing Computer Processed Accounting Records," Journal of Accountancy (October 1975), pp 46-59.

33. A. Milton Jenkins and Ron Weber, "Using DBMS Software as an Audit Tool: The Issue of Independence," Journal of Accountancy (April 1976), pp 67-69.

34. Gene H. Kiefer, "Systems Auditing with Test Decks," Management Accounting (June 1972), pp 14-18.

35. J. Gregory Kunkel, "Continuous Auditing by Exception," Management Accounting (July 1974), pp 45-48.

36. E. B. Levine, "Auditing Requirements for Advanced Systems," Journal of Accountancy (March 1974), pp 74-76.

37. Charles R. Litecky and Ron Weber, "The Demise of Generalized Audit Software Packages," Journal of Accountancy (November 1974), pp 45-48.

38. W. C. Mair, K. W. Davis, and D. R. Wood, "Parallel Simulation -- A Technique for Effective Verification of Computer Programs," EDPACS (April 1975).

39. Edward M. Milko, "Auditing Through the Computer or Around?" Management Accounting (August 1970), pp 45-48.

40. William E. Perry and Henry C. Warner, "Systems Auditability: Friend or Foe?" Journal of Accountancy (February 1978), pp 52-60.

41. W. Thomas Porter, EDP Controls and Auditing (Belmont, California: Wadsworth, 1974).

42. Robert Randall, "Computer Fraud: A Growing Problem," Management Accounting (April 1978), pp 61-64.

43. Hal J. Reneau, "Auditing in a Data Base Environment," Journal of Accountancy (December 1977), pp 59-65.

44. Larry E. Rittenberg and Gordon B. Davis, "The Roles of Internal and External Auditors in Auditing EDP Systems," Journal of Accountancy (December 1977), pp 51-58.

45. George H. Rittersbach and Stephen D. Harlan, Jr., "Auditing Advanced Systems," Journal of Accountancy (June 1974), pp 83-85.

46. Carol A. Schaller, "Auditing and Job Accounting Data," Journal of Accountancy (May 1976), pp 36-42.

47. William J. Schroeder, "Memo on Installing a Data Base System," Journal of Accountancy (February 1974), pp 81-82.

48. George M. Scott, "Auditing Data Base," CA Magazine (October 1978), pp 52-59.

49. Clarence O. Smith and Geraldine J. Jasper, "Using the Computer in Audit Work," International Accountant (October 1973), pp 8-12.

50. Stanford Research Institute, Systems Auditibility & Control Study, Vols. 1, 2, and 3 (Altamonte Springs, Florida: Institute of Internal Auditors, 1977).

51. Robert L. Stone, "Who Is Responsible for Computer Fraud?" Journal of Accountancy (February 1975), pp 35-39.

52. Michael W. Stoneham, "Data Base Auditing - A New Complexity," CA Magazine (May 1979), pp 32-38.

53. D. A. Thomas, B. Pagurek and R. J. Buhr, "Validation Alogrithms for Painter Values in DBTG Databases," ACM Transactions on Database Systems (December 1977), pp 352-369.

54. J. S. M. Verhofstad, "Recovery Techniques for Database Systems," Computing Surveys (June 1978), pp 167-195.

55. Joe Wasserman, "Selecting a Computer Audit Package," Journal of Accountancy (April 1974), pp 30-34.

56. Ron Weber, "An Audit Perspective of Operating System Security," Journal of Accountancy (September 1975), pp 97-100.

57. H. L. Weiss, "The Use of Audit Indication and Integrated Test Facility Techniques," EDPACS (July 1973), pp 8-11.