Developing A Comprehensive Curriculum For Accounting Information Systems: A Model-Oriented, Tool-Enhanced Approach

Joseph H. Callaghan, (e-mail: callagha@oakland.edu), Oakland University
Thomas W. Lauer, (e-mail: lauer@oakland.edu), Oakland University
Eileen Peacock, (e-mail: peacock@oakland.edu), Oakland University

Abstract

This paper describes a comprehensive approach to teaching Accounting Information Systems (AIS), outlining a curriculum that spans three courses. The curriculum uses a Model-Oriented, Tool-Enhanced (MOTE) approach and follows the systems development life cycle. The Accounting Education Change Commission (AECC) has proposed taking a wider view of accounting information and incorporating a systems perspective of the enterprise into the curriculum. In addition, the International Federation of Accountants recommends improvement in the information technology components of the accounting curriculum. Our new curriculum effectuates AECC proposals, incorporates some aspects of IFAC guidelines, and provides the opportunity for an integrated, interdisciplinary sequence of courses. The three-course sequence trains students in the planning, analysis, design and implementation of business systems.

The Need for Change

The need for curricular change derives from five areas: 1) the necessity to address the AICPA's amended bylaws regarding educational requirements, 2) the need for the professions to respond to information technology advances, 3) statements made by the Accounting Education Change Commission regarding the inclusion of design and use of information systems, and 4) the current disparate way of handling Accounting Information Systems (AIS) courses in the accounting curriculum, 5) IFAC guidelines (IFAC, 1996) regarding technology skills of accountants.

The advent of the AICPA's amended bylaws, requiring new members after the year 2000 to have 150 semester hours of education,

Readers with comments or questions are encouraged to contact the authors via e-mail.

has generated much activity and discussion regarding effective ways to deal with this rule change. Many schools have turned to offering a 5-year curriculum leading to a master's degree in order to encourage students to stay in school to complete the additional requirements. This change in requirements has created a challenge for educators. The issue is how best to structure the additional requirements to comply with students' need to learn skills that enhance their career prospects, the profession's needs for certification and skills, and the institution's need to satisfy its mission. The revised curriculum presented in this paper offers an alternative which is designed to give the accounting student the skills required to be involved in the planning, analysis, design, and implementation of accounting information systems. It offers the profession an employee who has many more diverse skills than the traditional accounting graduate. It offers the institution the ability to be on the leading edge of providing graduates who can enter the workforce more equipped to face real world problems.

The 1996 International Federation of Accountants (IFAC, 1996) issued International Education Guideline No. 11, *Information Technology In The Accounting Curriculum,* to provide guidance to member bodies in developing programs to enhance the competence of their members in the use of information technology (IT). In particular the guideline identifies the IT educational requirements for professional accountants under five main headings: *General* IT education requirements, the accountant as a *user* of IT, as a *manager* of information systems, as a *designer* of business systems, and as an *evaluator* of information systems. The guideline assumes that all professional accountants should be "proficient in the *general* and *user* educational requirements, as well as at least one of the other three roles." The proposed curriculum provides many of the educational requirements needed to comply with this statement. It does provide the opportunity for the student to become proficient in many of the tasks required to be identified with the *designer* role. In doing that, one can assume that it also provides the skills for the *manager* and *evaluator* role.

In 1990 the AECC in their Position Statement No 1 (AECC, 1990) state that " Accounting courses should present accounting as an information development and communication process." In addition, they promulgate the viewpoint that the essential components of courses should include the design and use of information systems. This theme is followed up in the AECC Position Statement No. 2 (AECC, 1991). This statement suggests that the primary objective of the first course in accounting is for students "to learn about accounting as an information development and communication function that supports economic decision-making" and that after completion of the first course it is expected that the students should understand "the basic features of accounting and reporting by organization, including the principles underlying the design, integrity, and effectiveness of accounting information systems." Unfortunately, most principal courses continue to emphasis a debit-credit, manual systems approach that is barely related to how AIS are actually developed in the current computerized, database environment.

These statements from the AECC highlight the need for a systems approach in teaching accounting. This systems approach would presumably emphasize a higher level view of the AIS, rather than a technical, parochial view. This paper presents a new curricular approach to teaching AIS that reflects this emphasis by incorporating into the constituent courses the teaching and learning of more abstract data and activity modeling skills that are technologically independent of any particular target system. Ironically, it is the development of technology that permits this abstraction from both academic and practice's specific technological implementations of AIS. This technologically independent, more abstract approach would reflect the wishes of the AECC in providing students with a sense that accounting can be thought of as "an

information development and communication function that supports economic decision making."

Recent studies (Davis and Leitch, 1988, Groomer and Murphy, 1996, Macur, 1998) have demonstrated the diversity of content in AIS courses. There are a variety of textbooks available for the AIS course. These predominantly take a "traditional" approach, that presents descriptions of information systems and how accounting systems require some unique knowledge. These books are used in the "traditional" AIS course where students learn how accounting systems differ from "other" information systems and complete some project related to some technology, such as spreadsheets or databases. The textbooks often cover a vast array of topics, allowing the instructor to choose a favored group. Borthick (1996) reiterates the problem of diversity of approaches in the AIS course, noting that the rapid change of technology further aggravates the problem.

Many other accounting courses, in which there is wide agreement as to both the content and conduct of the courses, can be characterized by the lack of differentiation in the teaching methods employed. For example, Intermediate Financial Accounting content is typically broken into two semester courses, the first course covering asset accounting, with the second course covering liability and equity accounting. Both courses usually emphasize rules reflected by evolving generally accepted accounting principles (GAAP). A lecture-discussion-problem-test sequence typifies the conduct of those courses. On the other hand, the content and conduct of AIS courses reflect the lack of consensus both, on what should be taught and how it should be taught. Course content ranges from workstation skills, such as word-processing, spreadsheet, and database skills, to theoretical systems approaches, with an emphasis on diagramming techniques (e.g. systems and document flowcharts) associated with traditional accounting cycles. Orientations can range from a managerial emphasis to an auditing emphasis. Moreover, the conduct of the courses can range from a conventional lecture-discussion approach to a computer projects-based approach.

This lack of consensus can probably be attributed to the lack of standard prerequisites for the AIS course, general disagreement about what should be taught, varied instructor backgrounds, varied and competing software solution approaches, and the diverse technical environments available at the institutions offering the AIS course.

One final variable, however, needs to be discussed, namely how various courses and accompanying textbooks bundle software to facilitate content coverage and the extent to which the texts support a computer-project orientation to the conduct of the course. Textbooks that do not bundle software often include spreadsheet templates and rely on the institution to provide the necessary related software, e.g. Microsoft Excel™. Textbooks that do bundle software reflect the author's anticipated approach to the conduct of the course. Thus, for a workstation skills course, an integrated software package, e.g. Microsoft Office™, might be included. For courses that aim for student exposure to commercial accounting solutions, a general ledger package, e.g. Computer Associates' ACCPAC Plus™, might be employed. For more general approaches, the textbook might bundle a relational database package, such as Borland Paradox™. Finally, for programming approaches to AIS, a higher-level language, e.g. Microsoft C++™, might be adopted. Therefore, textbook adoption may lead to an institutional commitment to the various software approaches, perhaps requiring coordination with pre- and post- requisite courses offered by other departments, if not other schools, within the institution.

The course content (and accompanying software adoption) decision will likely be affected by the supporting technical environment of the institution. Operating system choice, the

availability and type of networks, mainframe versus mini or micro computer hardware, and current software availability are issues that may need to be addressed. Operating systems might include MS-DOS™, Windows 95™, UNIX, VMS™, or other popular systems. In addition, the decision may be further complicated by the hardware environment and the type of network available for academic computing. Given the software bundling issue already discussed, the existing software environment may impact course content choices. In short, in the current situation, the AIS course content decision may be affected by operating system, hardware, and software choices already committed to by the institution. Thus, apart from important issues such as textbook quality, content coverage, and ancillary textbook support materials, the course content decision may be driven by other factors.

The New Curriculum

The overall goal of most AIS courses is to provide the student both a conceptual/logical framework to analyze accounting systems and an understanding of the practical/physical-level implementations of these systems. At the conceptual level, students are exposed to methods that permit abstraction of AIS, including diagramming techniques and programming concepts. At the heart of these methods is the ability to model accounting data, activities and their interactions, without getting into particular implementation strategies. This enables the student to understand the broad range of business issues addressed by AIS. On the practical/physical-level end, the abstract methods are then applied to particular environments. The choice of application includes: spreadsheets, databases, general ledger packages, and particular programming languages, all implemented in particular computing environments. Ideally, in addition to the specific technology, the student learns how accounting cycles are implemented in a specific hardware/software environment.

One way of describing the range of treatments of AIS courses is in terms of the relative emphasis of the conceptual and practical. Emphasizing the conceptual at the expense of the practical leaves the student without adequate grounding to understand how everything fits together. Conversely, a bias toward the practical usually implies settling on a particular choice of implementation. The student is left with a set of skills in the absence of a conceptual context that will not transfer well to other environments. Usually, due to the enormity of the task of including judicious coverage of both the conceptual and the practical, compromises are made.

Given the various treatments of the content and the conduct of AIS courses, the various approaches currently taken by AIS textbook authors, and the current technical environments supporting AIS courses, new software developments afford an opportunity to overcome the current tangled context in which AIS courses' content decisions are made.

The revised curriculum consists of three related courses: AIS, Design and Construction of AIS, and AIS-Special Applications. These courses are designed to encompass the four stages of the systems development life cycle: Planning, Analysis, Design and Construction. The first course includes planning and analysis, the second, design and construction, and the third is a complete application of the entire cycle, using a realistic scenario.

A disciplined, normative approach to systems development, information engineering (IE), and software development tools used to support the IE framework, notably program development or Computer-Aided Systems Engineering (CASE) tools, are key enablers to the MOTE approach we take to the conduct and structure of the AIS course sequence. MOTE stands for **M**odel **O**riented **T**ool **E**nhanced.

Model Oriented means that a conceptual foundation of the enterprise is developed that is based on models of data, processes, and their

interactions. Tool Enhanced refers to the use of an appropriate software development tool to support information systems modeling and the construction and implementation of actual accounting information systems. This takes advantage of a trend during the last several years in information technology, namely to provide professional programmers with a variety of software development tools that facilitate the production of higher quality applications in less time. These tools provide support for analysis, design, and construction activities and are characterized by technological independence, i.e. independence from specific IT environments, user system interface, database management systems, hardware, operating systems, and networks. The MOTE approach makes it possible to provide an even balance of the conceptual and the practical by leveraging the student's abstraction skills, while concurrently allowing the construction and implementation of actual accounting information systems.

Since information engineering and its dependence on software development tools is fundamental to the rationale for developing a new approach to AIS, the following presents a brief overview. IE, as described by James Martin, (Martin, 1988) consists of a set of structured techniques to be applied on an enterprise-wide basis or to a large segment of an enterprise. Based on the recognition of the strategic importance of the information resource for an organization, it progresses, top-down, through four stages: Information Strategy Planning, Business Area Analysis, System Design, and Construction. IE creates a framework for developing a suite of application information systems in an enterprise by using software development tools to build data models, process models, and system designs and to store them in a repository.

The First Course: Accounting Information Systems (this course is required for the accounting major)

This course has three core components. Part I is a general introduction to information and accounting information systems (generally manual systems). In the first part students review the financial accounting cycle, general ledger systems, general journal and special journals. The Input-Process-Storage-Output (IPSO) framework is established for modeling the enterprise as a system. Following the Hollander, *et. al.*, (Hollander, 1996) implementation of McCarthy's Resource Event Agent (REA) (McCarthy, 1979, 1982)modeling approach to AIS, the enterprise is decomposed into business processes. For each business process, the business events of interest are identified and REA modeling is used to model these events. These business processes roughly correspond to the traditional accounting cycles, such as Sales/Account Receivable/Cash Receipt, Factor Acquisition/Accounts Payable/Cash Disbursement, and Job Cost (Conversion). The data and activity models implied by the foregoing are documented by students in their first computer project, based upon a business scenario developed for this course. This project corresponds to the planning phase of systems development.

Part II looks at Accounting Information Systems Development, introduces the Systems Development Life Cycle and program development tool support, teaching the IE framework and the four phases of systems development: Planning, Analysis, External Design, Internal Design, Construction. The technical environment, hardware and software are introduced. Finally in this second part of the course the Relational Model and Programming Concepts, Data Modeling, Activity Modeling and Interaction Analysis are introduced.

Part III, the final section, further examines one of the business processes identified in the first computer project. This second project corresponds to the analysis phase of systems development. A completely developed data model is documented and is fully represented by an Entity Relationship Diagram (ERD), using the data modeling feature of the program develop-

ment software. An activity model is developed for a selected business process and documented using both the Activity Hierarchy Diagram (AHD) and an Activity Dependency Diagram (ADD) features. This method of process modeling hierarchically decomposes the higher level processes into elementary processes and then shows dependencies between the elementary processes. Finally, Process Action Diagrams (PADs) are developed to outline the pseudo logic of the elementary process, further confirming both the data and activity models.

The Second Course: Design of Accounting Information Systems (this is an elective or second course in AIS minor)

System Design is concerned with describing and documenting how specific procedures are used to implement the business processes. The design of procedures leads directly to their implementation in the Construction stage through the generation of computer code, user interfaces, and the creation of a relational database.

This course has three parts. In Part I, a review of the systems development life cycle through the analysis phase is conducted. A selected business function is given to the student in the form of an ERD, an AHD, an ADD and related PADs. User interfaces, in particular graphical user interfaces (GUIs), are discussed and developed, along with various processing methods and topologies. More specifically, batch, on-line, and real-time processing methods are explored; and client-sever environments are examined.

Part II entails a project that designs a given business process by establishing the Procedure Action Diagrams (PrADs) and Dialog Flow Diagrams. Whereas PADs document *what* needs to be done by the system, the PRADs establish *how* the system is implemented, given a processing style and interface choice. Window designs are established that permit the system user to interact with action modules and the underlying data model. The dialog flows establish the pathways the user will follow in interacting with the system components.

Part III of the course relates to how an actual system is generated in a particular computer environment. An operating system, a higher-level programming language, and relational database are selected for construction. The PADs and PrADs are code generated into the chosen language. The data model is converted into the chosen relational database. The GUIs are generated for the chosen operating system. This completes the systems development life cycle to produce an actual running application for the business systems selected.

The Third Course: Special Application (an elective or third course in AIS minor)

The objective of this course is to take the student through an exercise where they use all they have learned in the first two courses. They will plan, analyze, design and implement a business system. In order to do this, the course uses a sophisticated business scenario that introduces the student to problems related to the allocation of overhead in a complex manufacturing environment. One such program development would be American Tire Manufacturing: Applying Activity Based Management/Costing (Albright and Samson, 1993). Another program development, which might be more suitable for graduate classes, could be Schrader Bellows, a Strategic Cost Management Program development (Cooper, 1986). Both program developments are concerned with the determination of product cost and product profitability. The program developments involve manufacturing companies with a range of products (Schrader Bellows is significantly more complex), and several different processes or machines. In both program developments the companies, are experiencing economic difficulties. Both program developments lead the students into an overhead allocation system that is activity based.

The first part of the course will be the solution of the program development. The raw data is given in a spreadsheet format. Students must manipulate this data in order to find the appropriate solution (in these program developments, appropriate cost drivers). In the second part of the course, the students would model the redesigned system.

Benefits

This curriculum design fits well with the recommendations laid out by IFAC, Guideline No. 11. The first AIS course along with the pre-core and core Management Information Systems (MIS) classes and the computer applications required in the majority of major accounting classes, provide that the student becomes "proficient in the *general* and *user* educational requirements ...." The second and third course offers many of the skills required for the specialized role, that of *designer.* Also in that regard, topics taught over the three course sequence cover the role of information in organization design and behavior, system design techniques, and system acquisition: development life cycle phases, tasks, and practices and maintaining control over system development processes.

This three course sequence forms the basis of any of the following: a systems track with an accounting major; an accounting systems minor for those in other major areas, such as MIS, Finance, and Marketing, etc.; and a systems track within a 150 credit hour program or Masters program.

From a pedagogical standpoint, a highly integrated program development tool facilitates a leveraging of students' higher-level planning and analysis skills, with much less emphasis of technology dependent skills, such as detailed knowledge of particular operating systems, relational database management systems (RDBMS) or programming languages, like COBOL or C++. One commercial implementation of a highly integrated program development tool is Sterling Software's COOL:Gen™. COOL:Gen™ supports construction of information systems for a variety of operating system, RDBMS and programming language combinations.

The planning, analysis, and external design phases of information systems development are technologically independent, leaving only internal design and construction to specify particular technology combinations. Further, COOL:Gen ™ itself has been implemented for a variety of operating systems, including OS2™, Windows 95™, UNIX and Windows NT™. Thus, with the use of this integrated program development tool in AIS courses, there is much less concern about an institution's particular technical environment.

More importantly, students acquire both higher level conceptual skills and practical skills that will be useful in a wide range of technical environments. Moreover, the instructor need not struggle over the traditional tradeoff between teaching diagramming skills or programming skills. With an integrated program development tool, diagramming skills are seamlessly connected to programming skills. Finally, the implications for the conduct and content of AIS courses are clear. A radically new approach is available those willing to adopt a MOTE framework.

Given the advantages of this model-oriented, tool-enhanced approach, AIS courses that incorporate these advantages, by bundling a student version of a software development tool, e.g. COOL:Gen™, would fulfill the need for a more general and uniform approach to teaching AIS. The transition of general information systems development to accounting information systems development is relatively straightforward. After all, AIS were among the first business systems to be computerized and they constitute a major subset of management information systems. This is not to say that there are no uniquely accounting-related issues involved. AIS can be characterized as mission-critical,

transaction-based, and high-volume systems. This, in turn, requires a particular accounting orientation in developing these systems. In particular, the auditability and related internal control aspects of these systems would need special emphasis. Additionally, especially given the manual system orientation of most accounting students entering an initial AIS course, attention would need to be paid to demonstrating the computerized, relational approach to AIS development. This, in turn, would require that students to translate their debit and credit analysis methods to the normalized, table approach methods of the relational model. Finally, since the conduct of the courses is projects-based, the construction of relevant accounting scenarios is an integral part of the redesigned AIS curriculum.

Suggestions for Further Research

There are two aspects to further research: the relevance to practice and need to address completeness of content. Although traditionally the academic world leads practice, in the systems area, the high tech initiatives and their implementation are usually driven from the practitioner side. Therefore, practitioners in the public and private sectors of practice should review the curricula. Their systematic feedback would ensure continued relevance to practice. In addition, an important consideration for any pedagogical approach is whether students are able to learn and apply important concepts. Research should be conducted for this type of assessment. In terms of content, the areas of object-oriented modeling or development for internet-based applications have not been included in this current innovation. Expanding modeling-orientation to include a Unified Modeling Language (UML) or other object-oriented approach to modeling could include these areas. It appears that business databases are evolving along these lines and that relevant curricula should incorporate these concepts. Several program development suites have begun to incorporate tools to address these issues, for example Sterling Software's collection of COOL™ tools and Powerbuilder. Included here is the integration of object-oriented programming and object-oriented data modeling techniques that more formally integrate activity and data modeling. Finally, a comprehensive comparison of model curricula with the IFAC recommendations is desirable in order to ensure that the curricula is comprehensive enough to support the many roles that accountants play in the AIS arena. 🕮

References

1. Accounting Education Change Commission. *Objectives of Education for Accountants, Position Statement No. 1.* (1990).
2. Accounting Education Change Commission. *The First Course in Accounting, Position Statement No. Two* (1991).
3. Albright, T. L. and Samson, W. S. *American Tire Manufacturing: Applying Activity Based Management/Costing.* AICPA Program Development Program. Program development No. 93-07. 1993.
4. Borthick, F. "Helping Accountants Learn To Get the Information Managers Want: The Role of the Accounting Information Systems Course." Editorial. *Journal of Information Systems* 1996. Volume 10. No. 2, Pp. 75-86. 1996.
5. Cooper, Robin. *Schrader Bellows, a Strategic Cost Analysis.* Harvard Business Software Module. 1986.
6. Davis, J. R., and Leitch, R. A. "Accounting Information Systems Courses and Curricula: New Perspectives". *Journal of Information Systems* (Fall): 153-166. 1988.
7. Groomer. M. S. and Murthy,U. S "An Empirical Analysis of the Accounting Information Systems Course. *Journal of Information Systems.* Volume 10, No. 2, Pp. 103-127. 1996
8. Hollander, A. S., E. L. Denna, J. O. Cherrington, *Accounting, Information Technology, and Business Solutions*, (Irwin, 1996).
9. International Federation of Accountants

(1996) International Education Guideline No. 11, *Information Technology In The Accounting Curriculum*

10. Macur, K. M. "The Curricular Content of Accounting Information Systems". *Review of Accounting Information Systems*. Volume 2, Number 3. pp. 11-19. Summer, 1998.
11. Martin, J., McClure, C., *Structured Techniques: The Basis for CASE*, (Prentice Hall, 1988).
12. McCarthy, W. E., "An Entity-Relationship View of Accounting Models", *The Accounting Review* (October 1979), pp. 667-686
13. McCarthy, W .E. "The REA Accounting Model: A Generalized Framework for Accounting Systems in a Shared Data Environment", *The Accounting Review* (July 1982), pp. 554-578.