EDITORIAL

Helping Accountants Learn to Get the Information Managers Want: The Role of the Accounting Information Systems Course

A. Faye Borthick
Georgia State University

Before they can get the information they want, users must know whether the potential information could or does exist and be able to get it.1 Users are not born knowing how to do this—they must learn to do so. Historically, undergraduate accounting curricula with courses in financial accounting, managerial accounting and taxation make students aware of the potential existence and availability of traditional kinds of financial information. Accounting academicians and professionals generally agree that the accounting information systems (AIS) course should help students become proficient with information systems (AAA 1986, 1987; Perspectives 1989; AECC 1990; IFAC 1995; AICPA 1996). Accounting organizations also agree that accountants have roles both as users of information technology and as managers, designers and evaluators of information systems (IFAC 1995; AICPA 1996). The unanimity disappears, however, at the point of specifying exactly what it means to be “sufficiently proficient with information systems” and how to inculcate that proficiency.

This editorial proposes a set of learning objectives that define systems proficiency for accounting students. Accompanying the learning objectives are explanations of activities that students could undertake to develop the competencies that the learning objectives imply.

The lack of unanimity in AIS course objectives is apparent in the diversity of topical coverage and computing-related projects in courses offered at different institutions (AAA 1987; Davis and Leitch 1988; Groomer and Murthy 1996) and in the wide variation of the placement of the AIS course in the curricula of different institutions. This diversity is likely a function of five factors: the low proportion of AIS faculty actually trained in AIS (Campbell 1987), the heterogeneity of students’ information systems backgrounds, rapidly changing computing environments, the difficulty of

1 Borthick (1992) made the case for a technology-enabled shift from a world characterized by two-person information economics, in which an evaluator selects the information system and a separate person (a user or a decision maker) uses the information system’s output to select actions, to one characterized by a single-person model, in which the same person selects the information system and subsequent actions.

The editor is indebted to Ron Clark, Bill Cummings, Anita Hollander, Bill McCarthy, Uday Murthy, Dan Stone, Jim Scheiner, Gary Schneider, and participants at the 1996 Federation of Schools of Accountancy Directors of Masters Programs Conference for helpful comments on earlier versions.
maintaining adequate institutional computing facilities, and the inertia engendered by traditional teaching of accounting. Except for the last one, these factors are unique to the AIS course. The rapidly evolving roles of accountants as users of information technology and as designers, managers and evaluators of information systems means that the need for systems expertise is expanding—which further increases the difficulty of keeping the AIS course current.

The lack of unanimity in AIS course objectives affects AIS faculty. They are often the only AIS person at their institution, and some of them are junior faculty trying to establish research programs. These faculty find themselves also undertaking substantial curriculum development to prepare students for careers in which accounting and technology are synonymous. Even if doctoral programs prepared them well for these atypical early-career dual roles of researcher and curriculum developer, junior AIS faculty may feel that their time is better invested in their research programs. The scarcity of AIS-trained faculty also results in senior faculty trained in other areas teaching the AIS course. Although they may have established research programs, these faculty are often unprepared to develop AIS curricula.

Regardless of how the diversity in AIS course content arose, there is a compelling argument for approaches that help students gain more information systems expertise. If increased competence improves one's productive capacity in information-intensive environments, then the more competence students can develop in the AIS course, the better off they will be as job candidates and employees (Elliott 1992; McKinnon and Bruns 1992; Cooper 1996a, 1996b). Managers will increasingly need accountants that can obtain information from disparate sources and analyze it in useful ways. Information systems expertise will help accountants be more productive in any area of accounting regardless of whether they have four or five years of accounting education. Better prepared professionals can improve the functioning of their organizations and, through them, society in general.

For tomorrow's environment, a single undergraduate AIS course is insufficient. Because using information technology effectively requires practice in context, use of information technology should be integrated throughout the accounting curriculum. For example, student development of models with analysis tools such as spreadsheets and statistical analysis software is appropriate in (1) cost/managerial courses for analyzing costing alternatives and preparing budgets; (2) financial courses for determining the effects of different accounting methods; (3) taxation courses for tax planning; and (4) auditing courses for performing analytical review. Ideally, the concept of accounting as an information system permeates all the accounting courses such that each course builds on the systems competence developed in earlier courses. Regardless of the placement of the AIS course, early in the curriculum as a foundation course or after other courses impart specific accounting precepts, the need for integration of information systems in all accounting courses is the same.

Consistent with other accounting specialties such as taxation, a master's degree is increasingly the entry-level educational preparation for AIS. But it is unrealistic to think that acquiring AIS competence can be delayed until the fifth year of accounting education. First, one year is simply not enough time for students to develop systems expertise. Second, students elect a master's degree because they want to specialize in some aspect of accounting. If they have not had a substantial undergraduate AIS experience, they will be unaware of the potential benefits of an AIS specialization. Masters' specializations build on the undergraduate AIS course, which should include a survey of AIS content and give students practice in thinking with computer tools to solve current problems and respond to business opportunities.

**LEARNING OBJECTIVES FOR THE AIS COURSE**

Accounting professionals need systems proficiency as users of information technology and as designers, managers and evaluators of information systems. I propose that students can attain systems proficiency by developing competence in the categories of information use, documentation, data modeling, system development, and internal control as outlined and discussed below:
I. Information use
   A. Decide what information would be relevant to solving a business problem.
   B. Extract or create the needed information.
   C. Analyze the information to solve the problem.

II. Documentation
   A. Understand symbolic representations of processes.
   B. Modify and create symbolic representations, e.g., processes that:
      1. Update files.
      2. Conduct business via electronic commerce.

III. Data modeling
   A. Identify and model business processes.
   B. Develop data models that correspond to business process models.

IV. System development
   A. Design and implement systems with appropriate data models for routine tasks.
   B. Propose, design and implement systems with appropriate data models to solve innovative tasks.

V. Internal control
   A. Evaluate internal control in information systems.
   B. Design and implement internal controls in information systems.

Students that achieve these learning objectives should be well-equipped to provide managers the information they need rather than just generating traditional reports (Burton 1984; Cooper 1996a, 1996b; Rohan 1996; Reeb and Cameron 1996). Managers do not care that the information they need to be competitive may be fragmented across different computers running different operating systems in different time zones. They just know they need the information to run their businesses. Good managers heed Drucker’s (1992, A16) admonition to ask themselves “What information do I need to do my job? When do I need it? In what form?” Accounting professionals that achieve these learning objectives as students would be ready to help such managers.

This is an ambitious agenda, but organizations now demand more than ever from employees. If non-accountants learn to respond better than accountants to managers’ needs for information, then accountants will become less valuable to organizations (Cooper 1996a, 1996b). Non-accountants are certainly capable of developing insightful analyses of quantitative and qualitative data from disparate sources to guide organizations in new and continuing circumstances. Managers are still relying on accountants to interpret the meaning of financial information and to ensure that the information is reliable. This creates an opportunity for AIS faculty members to prepare accounting students for careers in information-intensive environments—environments in which creative analysis is more valuable than encyclopedic knowledge of accounting procedures (Siegel 1996).

Although it is the most likely home for the learning outlined above, the AIS course is not the only instructional configuration for imparting these learning objectives. For example, students could learn about documentation and system development from courses in analysis/design and programming taught by departments of information systems. Internal control matters could be taught in auditing courses, and the use of information systems to solve accounting problems could be woven throughout other accounting courses. There are, however, potential costs to fragmenting the learning objectives across multiple courses. Even if all the individual learning objectives were covered sufficiently, students might not realize the implications of each learning objective for the others. For example, if students learn about system development from an analysis and design course in a department of information systems and learn about internal control in an auditing course, it is unlikely that students would think of internal control as a function of system design. In the past, accounting systems relied on manual reconciliation controls. In those systems, internal control design was not so crucial because any lapse in control could be remedied with another manual procedure. In highly automated systems—the ones with no paper documents whatsoever—continuous control and monitoring...
must be a function of design. Students will better realize the importance of internal control as a function of design if they learn about system design and internal control together. Similarly, students will better learn to solve problems that cross organizational boundaries if the problem solving does not occur only in the context of courses that focus on specific areas of accounting, e.g., financial or managerial accounting. In the AIS course, students could solve problems that require them to transcend organizational boundaries and traditional approaches to recurring problems.

INSTANTIATION OF LEARNING OBJECTIVES
The proof of learning is in its instantiation—what students learn to do while engaged in learning activities. This section explains examples of activities that define each learning objective. These activities are very much in the spirit of intentional learning because they depend upon students learning skills rather than automatically acquiring them (Francis et al. 1995). These activities create opportunities for students to be independent learners that develop competency in assimilating and analyzing complex, technical materials, in seeking out information and developing solutions to business problems, and in distinguishing between relevant and irrelevant material (Francis et al. 1995).

I. Information Use
A. Decide What Information Would Be Relevant to Solving a Business Problem
As long as the technology of mainframe computers made data manipulation and storage expensive, “the constraining variable was the availability of data, not the need for it” (Burton 1984, A10). Before computer costs fell enough to permit ubiquitous computer use—symbolized by a personal computer (PC) on every professional’s desk—accountants were spared the need to make many decisions about what data were uniquely relevant to a specific problem. Organizations could not afford to keep much data in usable form. The data they did keep were always aggregated in standard formats over arbitrary time periods. Now, organizations can have almost any data they want—they can collect internal data themselves and buy external data from others. Therein lies the problem for accountants—they now must decide which data they will bring to bear on particular problems. The arguments for better data choice to improve management and investment decisions are compelling—more insightful organizations prosper while those slow to apply creative approaches decline (Johnson and Kaplan 1991; McKinnon and Bruns 1992; Cooper 1996a, 1996b).

In one sense, accountants’ 50-year legacy of generally accepted accounting principles (GAAP) is an anachronism. GAAP specifies a small number of sanctioned data aggregations at a time when organizations increasingly value their managers’ ability to solve old and new problems in creative ways. Accountants’ accumulated experience with fixed-format reporting has likely automated their information choice and use decisions (Schneider and Shiffrin 1977; Schneider et al. 1984). This cognitive inertia (Reger and Palmer 1996) makes it hard for them to solve problems that require more than fixed-format results. Thus, an accountant trained to identify and solve problems by applying a few fixed-information formats will likely have more difficulty envisioning creative information uses than someone trained to think creatively about information gathering, aggregation, and use.

In traditional accounting programs, students enter AIS classes having been trained to approach accounting problems with a set of fixed-format reports that they studied in their financial and managerial accounting courses. Thus, requiring AIS students to decide what information they need to solve a problem will require their overcoming prior accounting courses in which they used small, complete data sets to solve problems. Despite curriculum redesign prompted by AECC (1990) grants, pervasive changes in accounting curricula are occurring slowly. Because curricula are so hard to change, AIS students are apt to be inculcated with a thorough grounding in fixed-format reporting in many accounting programs.

Prompting students to be creative in selecting data relevant to a business problem requires a different approach. Instead of working problems with small, complete data sets, students must work with imprecisely stated problems and only general guidelines for obtaining potentially relevant data. For example, students might explore hypertext files as they think about the problem. The hypertext
links would reveal more data—some relevant, some not—as students pursue the links. Some of the links could be to database tables and others could be to the World Wide Web. The problems presented to students should be manageable, growing in complexity and richness as students gained problem-solving experience. Working from imprecise problem statements is superb experience for entering a dynamic, chaotic world upon graduation.

B. Extract or Create the Needed Information

Having identified information relevant to defining or solving the problem, students would then extract or create the needed data. Data sources might include McKinnon and Brun's (1992, 222) "large real-time database[s] into which information is continually flowing" from operations and other sources, database tables that students create from other data, and files that students can electronically search. Software tools for data extraction could include database query languages such as SQL (Structured Query Language) and QBE (Query-By-Example) and Internet searching programs, all of which are readily available in PC versions.

C. Analyze Information to Solve Problems

Having identified and extracted relevant data, students are ready to analyze it to solve problems or respond to new opportunities. This kind of problem solving, known as "knowledge discovery" in databases and data mining (Fayyad et al. 1996), prompts students to think through what they are trying to do and what data they need. For example, a manufacturer might have noticed an unusually high failure rate for one of its products that is assembled from parts that come from several different sources. The failures do not become apparent until after final assembly. Given suitable data tables with information on raw material source, part fabrication, product assembly and product failure, students could investigate different hypotheses about the causes of the failures (Borthick and Roth 1997). They could, for example, trace the failures to specific parts made at specific plants or to inferior raw materials. This kind of problem is not routine, but managers face such problems every day and they want help finding solutions to them.

The product-failure problem illustrates ad hoc problem solving prompted by the need to remedy a condition—an abnormally high product-failure rate. Students must also learn to solve problems whose solutions would become the basis for continuous process monitoring. For example, companies using just-in-time (JIT) II, in which suppliers become responsible for the buying company's inventory replenishments, may want continuing assurance that suppliers are not taking advantage of them. Given suitable data tables, students could develop queries that could run automatically to detect suppliers that are not conforming to agreed-upon levels of stockouts, overstocks and defective items (Borthick et al. 1998).

A software tool for solving both of these problems—finding the cause of product failures and monitoring supplier performance—is a database query language. Its interface could be command driven, e.g., SQL, or data-view driven, e.g., QBE. Problem solving may also require use of a database programming language, e.g., when manipulations require that data be operated on one record at a time. When they become readily available for PC-based database systems, fourth-generation query languages and tools for on-line analytical processing (OLAP) (Ottolini 1996) would also be appropriate software for performing analyses.

As they extract and analyze information, students are likely to make errors such as misspecifying data and decision models, using inconsistent data, and omitting important variables (Ross 1996; Davis 1996). Part of students' learning to analyze information should include design principles that minimize the incidence of model errors (Ronen et al. 1989; Isakovitz et al. 1995) and evaluation techniques for detecting and correcting model errors (Borthick 1989).

Achieving sufficient software mastery will require substantial effort, especially if students' prior use of software tools has been limited to entering data into existing spreadsheet templates or performing single-table database operations. Software mastery develops over time as learners

---

2 Statistical analysis and audit command languages such as SAS and ACL are appropriate software tools for analyzing large volumes of data and conducting specific audit steps, but they are more likely to be used in more advanced AIS and auditing courses.
apply their skills to a variety of problems. An alternative to including software skill training in the AIS course is to make software proficiency a prerequisite to the course. By using proficiency tests (Stone et al. 1996), instructors can ensure that students have demonstrated the required skills before beginning the AIS course.

II. Documentation

A. Understand Symbolic Representations of Processes

Students need to understand symbolic representations of processes because such representations are the language practitioners use to portray systems behavior. To design, modify, manage or evaluate systems, one must be able to read and write the representation language. Understanding symbolic representations is also required to explain to others what a system can or cannot do. The most common representation schemes are flowcharts that portray processes implemented with specific technology and data flow diagrams that portray processes independent of technology-specific implementations. Most other representation schemes are combinations or variants of these two representation schemes.

B. Modify and Create Symbolic Representations

Students also must learn to create symbolic representations for new or modified systems. File updating and electronic commerce are both topics that require coverage somewhere in the AIS course, and both can provide experience in learning to modify and create symbolic representations.

1. Updating files. Using flowcharts comprises many students’ first encounter with sequential processes as implemented in single-processor computers. Students need to understand the nuances of file updating because it ensures the integrity of stored data. Sequential updating is typically the dominant processing mode in legacy systems, the decades-old mainframe-based systems that many organizations still use to maintain master and transaction files for accounting systems. Sequential updating is still important, however, because many transaction files are updated in batch sequential mode for the sake of efficiency, even in recently developed systems that use relational database technology. Updating indexed files is important because on-line processing in database systems is typically conducted with indexed files. For example, relational database systems usually store data in indexed files. An understanding of file updating can help students learn how to reconstruct failed processing sequences, and identify and correct the failure’s cause.

2. Conducting business via electronic commerce. Businesses increasingly engage in electronic commerce, conducted via fixed-format schemes such as electronic data interchange (EDI) and free-form formats on the Internet. Electronic commerce supports JIT II relations between trading partners in which suppliers assume responsibility for purchasers’ inventory levels. Using electronic commerce as a subject of system documentation promotes students’ understanding of how such systems work. Students would, for example, understand why trading partners might agree to dispense with traditional information exchanges such as invoices.

III. Data Modeling

A. Identify and Model Business Processes

Modeling business processes entails representing the salient aspects of business situations using one or more representation schemes, e.g., data flow diagrams/flowcharts, resources-events-agents (REA) (McCarthy 1979, 1982), or semantic object models (Codd 1976; Hammer and McLeod 1981). Modeling prompts students to perform a systematic analysis to identify what information giving

3 The information includes, but is not limited to, information concerning (1) what processes, (2) with what events involving what resources, (3) with which participants, (4) with what outcomes occurring when and where, and (5) with what results on whose behalf.
which facts is important enough to be recorded. Specific facts are important enough to be recorded if they, for example, support continuing business operations or permit required or potentially useful analyses. The modeling process shows students that they can identify and document business processes in multiple ways. Finding that there may be no single solution to a modeling task can frighten students accustomed to the closed-form procedures of traditional financial and managerial accounting courses. Students are often uncomfortable with the use of multiple models, no one of which is the best, but some of which are better representations for different purposes. Furthermore, because the business environment changes, the best representation for a specific purpose may change too. The revelation of multiple models, whose relevancy changes over time, does, however, show students that an organization's well-being depends on how perceptively it models business processes and how quickly it responds to changes that should prompt modifications to those models.

B. Develop Data Models that Correspond to Business Process Models

As they learn how to use database queries to solve problems, students become aware that the arrangement of data attributes matters. These arrangements of data attributes, collectively known as the data model, determine which facts are readily available to relate to other facts, the extent of redundancy in data tables, and the complexity and cost of retrieval and update queries. Whether they develop data models formally (Kent 1983) or informally, students need experience arranging data attributes in data tables to give the best possible model for specific uses.

IV. System Development
A. Design and Implement Systems with Appropriate Data Models for Routine Tasks

Without actually implementing systems, the idea of modeling is just an abstraction. The quality of data models emerges when students try to use them to create systems or solve problems. Therefore, students must implement enough system fragments that they become aware of how their data and process design choices affect the integrity and usability of systems. Given available software and computers, the most likely implementation platforms are database managers with programming languages on PCs.

Creating data models is quite challenging by itself. In my experience, accounting students are unable to cope with the intellectual demands of specifying a data model to a database manager and developing code that uses it if there is any ambiguity in what their process is to accomplish and in what order. Thus, students will have greater success if their first data model implementation is for a routine business process, e.g., basic purchasing functions, for which all actions have been specified in advance.

As they develop a segment of a processing system, students encounter another pervasive system phenomenon—that more potential features exist than they have the time or ability to implement. For example, PC database managers permit data attribute validations, e.g., table lookups for coded data values and range checks for dates, but each one has to be implemented. Through their implementation experience, students learn what organizations learn—that they must choose features carefully because resource constraints make it impossible to implement all possible desirable features.

B. Propose, Design and Implement Systems with Appropriate Data Models to Solve Innovative Tasks

Once they have become familiar with the database manager, e.g., by implementing a system for a routine task, students are ready to propose, analyze, design, program, test, and deploy a system for an innovative task. If the application concerns a business process that is an interesting subject in the business press, students can share their development experience with recruiters as evidence of their information systems competence and their creativity.

IV. Internal Control
A. Evaluate Internal Control in Information Systems

A philosophical difference distinguishes the approaches of accountants and information systems (IS) professionals to information systems. Because systems have failed in the past, accountants assume
that information systems have the potential to fail or be subverted. Thus, accountants believe that internal control can provide reasonable assurance that systems will meet their objectives. IS developers know that systems can fail, but the challenge of implementing systems on time and on budget usually takes priority over concerns about potential failures. Thus, IS developers tend to concentrate their energies on new functional requirements rather than internal control. Understanding this distinction between accountants' and IS developers' viewpoints helps students realize why systems might be implemented without adequate internal control.

Possible system failures do not create a license to impose absolute controls—quite the contrary. No one wants more control than is absolutely necessary. Controls make systems more expensive—and often less usable—because they require resources to implement, maintain, and operate. Therefore, students must learn how to evaluate internal control so they may design effective minimal sets of controls and identify controls on which they can rely in audits. Because controls in highly automated systems are so intricately embedded in the systems they control, the first AIS course is the ideal place for students to begin to learn to design and evaluate internal control in information systems.

The domain of internal control evaluation is broader than information systems per se. If business risk is “any event or action that stops an organization from achieving its stated goals or business objectives” (Villegas and Gates 1996, 6), then the growing integration of technology and business process makes the system assuror’s role more important. That is, as written records of transactions disappear and network functioning becomes synonymous with business activity, the evaluator becomes responsible for assessing the integrity of business processes. Regardless of whether an electronic audit trail exists, organizations will expect their auditors to provide ongoing assurance of business system functioning (Elliott 1994, 1995). Complicating the task of providing assurance services are organizations’ responses to new business situations that present equally new and not thoroughly understood control implications, e.g., conducting business over the Internet.

B. Design and Implement Internal Controls in Information Systems

Students can best develop their understanding of internal control by designing and implementing internal controls in systems they develop. Without this experience, they are unaware of design trade-offs, including the need to design and implement systems quickly. If students design and implement controls for their systems, they often see how a seemingly perfect design can result in an imperfect system. Changes in an organization or in its environment can prompt changes in requirements, which have the potential to make the original system design no longer a good fit with reality. Students, too, can experience the effects of changing requirements. Even for routine projects, but especially for projects students choose, students often realize that they will be unable to implement a system as originally specified because they cannot deliver the system in the time allotted with the software available to them. Thus, students, just like practitioners, change the requirements and the system to fit the available resources. Realizing the imperfectibility of information systems helps students develop professional skepticism toward systems.

OMITTED ACTIVITIES

Not every conceivable AIS-related learning objective can be included in one course. Therefore, to fulfill our role as educators, we must select the most crucial learning objectives for the AIS course and work with our non-AIS colleagues to ensure that the whole curriculum offers a series of progressively more meaningful experiences for the joint development of systems and accounting proficiency. The most obvious casualty of this opportunity-cost-of-learning is the encyclopedic mastery of facts not directly on the critical path of preparing students to be users of information technology and to be managers, designers and evaluators of information systems.

The role of trainers is to teach students “to deal with something they will encounter,” but our role as educators requires us to prepare students “to deal with anything they might encounter” (Mallach 1996, 37). For example, even though some employers might be receptive to graduates trained to install and use popular accounting software, such training has a short useful life. We shortchange students if
we do not insist they gain a broad understanding of information systems that enables them to work with any accounting software.

Because we are preparing accountants for 40-year careers in which user-information requirements and information technology will change many times, our responsibility is to ground students in concepts and principles that will ensure their ability to grow into a succession of jobs using evolving technologies. For example, as a result of developing their own system segments with a database manager and its programming language, students gain the systems maturity that enables them to install and use many different accounting software products. Although using proprietary accounting software—or seeing it demonstrated—can show students design deficiencies, explicit training in installing or using PC accounting packages is counterproductive. The time devoted to memorizing facts about today’s software products is much better spent developing capabilities to design and use any system. When they need training in specific software, e.g., to install or evaluate it for clients, accountants can get the training from the software vendor or independent training organizations. Product-specific training highlights the idiosyncrasies of particular software and can inhibit students from learning to think creatively about business processes. A better investment of learning time is to use software tools such as database management software or application development software to develop applications. The software development process reveals generally the behavior of software and gives students the systems understanding they need to use, design, manage and evaluate any information system.

RESEARCH OPPORTUNITIES TOO

A benefit of the AIS course containing learning activities that reflect what accounting professionals are expected to do is that the activities could inspire research in professional practice and education. Indeed, one test of course content appropriateness is to ask whether professionals perform the task. If professionals perform the task, it may be appropriate subject matter for students too; if professionals are not likely to perform the task, then the task may not be appropriate for AIS courses.

For business problem solving, the learning required of professionals and students may be similar. For example, one cannot search for information that supports or refutes possible causes of business or system phenomena without first generating hypotheses. Thus, professionals’ and students’ hypothesis-generation processes could be the subject of research into how experts develop hypotheses and how novices learn to be expert hypothesis generators. Faculty and students would be participants in a research partnership that could identify ways to accelerate students’ learning and improve professional practice.

THE FUTURE

As faculty shift the AIS course from an encyclopedic approach to one featuring active learning of tasks that professionals perform, more students will become intrigued with the rewarding careers they might have involving accounting information systems. Joint competence in accounting and information systems is increasingly necessary for successful performance of an accountant as a user of information technology and as a manager, designer and evaluator of information systems. More students will be interested in masters’ specializations in AIS. The core portion of such programs could include courses in analysis and design, database systems, data modeling, programming, object-oriented development, information systems auditing, management accounting systems, and information systems security and privacy. Elective courses could include, for example, process redesign, computer graphics, data structures, group decision making, decision support, project management, expert systems, multimedia systems, systems strategy, communications, process courses in statistical quality assurance, quality improvement, production, logistics, marketing, and finance, and industry courses in government, real estate, health care, international business, and others.

Masters’ graduates with dual competency in accounting and information systems are highly sought after for positions in consulting, auditing, management accounting, and system development. Acquiring sufficient depth in the master’s specialization is only possible if students experience enough information systems content in the undergraduate AIS course to realize they really could enjoy the creative, demanding, fulfilling work in an environment in which technology is inseparable from business process.
AN INVITATION

I invite you to respond to this editorial in the form of letters to the editor and proposals for the content of masters’ degrees specializing in AIS. This is a formative time in the evolution of such degrees and I am confident we will develop better degree programs with broad participation from the faculty that offer them and the professionals that want to hire AIS graduates.

REFERENCES


