Beyond the Model: Building an Effective and Dynamic IT Curriculum

Jeffrey Brewer, Alka Harriger, and John Mendonca
Computer and Information Technology Department
Purdue University, West Lafayette, IN, USA

jbrewer@purdue.edu, harrigea@purdue.edu, mendoncj@purdue.edu

Executive Summary

A model curriculum, such as that developed by the ACM/SIGITE Curriculum Committee (2005), has two important functions. First, it provides a base structure for newly developing programs that can use it as a platform for articulating a curriculum. Second, it offers an existing curriculum framework that can be used for validation by existing programs. The model does not, however, reflect many of the characteristics and considerations necessary to build a fully-functional, effective and dynamic curriculum. Necessary components for a comprehensive curriculum that are discussed in this paper include adaptation to the institution's mission and other program goals, responsiveness to local and regional needs, sensitivity to the availability of expertise and other technical resources, and a mechanism for response to a continuously changing environment.

This paper proposes a continuous improvement process framework for developing a comprehensive curriculum. The model consists of four phases: 1) Collect; 2) Evaluate; 3) Design; and 4) Implement. The Collect phase involves investigating current curricular standards, such as the SIGITE model curriculum, gathering data about the needs and desires of a program's stakeholders, and understanding the professional, technical and educational environment in which the curriculum will operate. The Evaluate phase integrates this data to develop a mission and set goals for the curriculum, define the outcomes expected of students, identify current and needed competencies, and specify other parameters that establish the foundation for an effective curriculum. The Design phase articulates the curriculum and assures that it is aligned with the goals, needs and resources identified. The Implement phase involves the establishment of the curriculum and a structure for governance and continuous evaluation and improvement. The cyclic nature of this model supports continuous improvement of the curriculum.

Any established curriculum must be dynamic enough to adapt to the constantly changing information technology environment. Educators and curriculum administrators who are charged with effective curriculum design and delivery must be aware of new technologies and develop strategies for incorporating them as needed. Collecting data about the curriculum's effectiveness, especially the satisfaction with stakeholders, begins another iteration of the cyclic process, with the objective of continuous improvement. The model curriculum provides a valuable starting point, but program developers need to use the process model to define and build an effective, dynamic curriculum that is responsive to their stakeholders' needs initially and throughout the life of the program.

Throughout this document, Purdue University's Computer and Information Technology department is used to describe specific components and techniques for implementing the proposed model. Other implementation techniques may appropriately be used to accomplish suggested goals of each model component.

Keywords: Information Technology, model curriculum, continuous improvement, process framework

Introduction

Efforts to define an Information Technology (IT) discipline across educational institutions began in earnest in 2001, and included parallel work to develop both IT accreditation criteria and a model IT curriculum. The Special Interest Group for Information Technology Education (SIGITE) of the Association for Computing Machinery (ACM) formed a special committee to write a formal IT curriculum document in 2003. The latest version of their report describes a core curriculum, which was developed from the input of more than 30 educators and professionals representing numerous educational institutions and organizations (ACM/SIGITE Curriculum Committee, 2005).

There is no doubt that the IT model curriculum document is a milestone in the continuing development of IT education. It is both a framework for curriculum development and a standard of excellence. A significant value of the model is that it provides an academically and professionally validated core body of educational content.

The ACM Joint Task Force for Computing Curricula (2005) also prepared an informative overview report that differentiates the various computing curricula, including IT, computer science, and information systems. Both the IT curriculum document and the overview report are effective tools for educational institutions considering the establishment of an IT program. For existing programs, they provide practical value for self-assessment and validation activities, and for preparation for the accreditation process.

However, a model such as that proposed by ACM/SIGITE is, by its nature, limited. As stated in the curriculum model document, the proposed body of knowledge "does not by itself constitute a curriculum" (ACM/SIGITE Curriculum Committee, 2005, p. 20). Regardless of its merit, the IT model is limited in its utility for developing a fully-functional, effective and dynamic curriculum. Although it presents examples of "elective" outcomes and suggests content and strategies for developing a curriculum, the model limits its scope to defining the core knowledge areas and knowledge units. This recommended core content constitutes less than 20% of the expected IT curriculum (ACM/SIGITE Curriculum Committee, 2005).

A good strategy for building a complete, effective and dynamic curriculum is to use the model as a base and to build on it and customize it to adapt to the needs and requirements of a specific program. As described in the model, a complete curriculum must include additional educational components such as fundamental knowledge in math, science, and interpersonal skills. Most programs will develop courses in specialized IT areas as well, to offer students deeper knowledge in particular areas. Some programs will choose specific application domains (such as in a particular industry) and may include interdisciplinary coverage in their curricula.

In addition to its intentional, limited scope, a second significant limitation of the curriculum model is that being a "standard," it is generic. As noted in the model document, a program must adapt to its own needs, circumstances and characteristics—including factors such as its mission and goals; the skills and knowledge of incoming students; faculty and laboratory resources; and job placement. For example, by virtue of its mission and funding, a program may have a regional focus—that is, its curriculum reflects the needs of the surrounding community it directly serves.

If there is a dominant technology, industry or employer, then it is appropriate to tailor a curriculum for that environment. Also, a program may elect to use the advantage of locally available faculty and professional expertise and other applicable resources to differentiate and distinguish its program.

A third limitation, perhaps the most significant, is that a curriculum model is relatively static. That is, it is a snapshot of the needs and expectations of a particular time. It is the product of an extended process in which ideas from many sources are gathered and examined by a broadly representative group of educators and professionals. If the revision timetable for this model is similar to those for other computing disciplines, it will take several years to update the model. Nevertheless, IT is a rapidly changing discipline. This may explain why the authoring committee itself recognized that "the curriculum must respond to rapid technical change" (ACM/SIGITE, 2005, p. 47) and recommended an established review process on a recurring basis.

Keeping up with information technologies and their profound impact on IT education is perhaps the greatest challenge to the establishment of an effective and dynamic curriculum. Programs that choose to incorporate emerging technologies will want to establish a mechanism for scrutinizing technology and performing continuous evaluation. For example, the "Information Assurance" body of knowledge in the model probably would not have been identified as a distinct core sub-discipline three or four years ago. Similarly, the Computer and Information Technology (CIT) program at Purdue University is developing courses and specialties in mobile and micro technologies, and interdisciplinary curriculum components in digital forensics, IT for healthcare delivery, and bio-informatics. These sub-disciplines of IT are not identified as part of the core, but are significant components of the future IT curriculum for CIT.

The proposed curriculum model is focused on a subset of a complete curriculum (the core body of IT knowledge), it is not specific to the needs of individual programs, and it is relatively static. Therefore, curriculum designers, planners and administrators need additional, supportive tools for establishing a complete curriculum. The proposed model is a process, not a product. Building a desirable curriculum requires a process that responds to the needs of stakeholders and reflects its own mission, resources, and the technical and social environment in which it operates. A curriculum model is a good starting point; but it is insufficient. This paper articulates a continuous improvement process framework that supports the creation of a functional, effective, complete and dynamic curriculum.

Prior to the implementation of this framework, the CIT department's curriculum development process used many of the same tools and techniques as those discussed in this paper, but it was not implemented in a formal continuous improvement framework. The process was ad hoc and intermittent. During the late 1990s and into 2000 many of the same tools and techniques were used to prepare for institutional accreditation under the North Central Association Commission on Accreditation and School Improvement (NCA CASI). In 2001, the Purdue University president instituted a campus wide strategic planning effort which started at the highest level of the university, then filtered down to the college level, and then finally to the department level. The strategic planning effort for a short time diverted attention away from the formalized curriculum development and assessment process.

A Framework for Developing a Complete IT Curriculum

The authors advocate the use of a continuous improvement process framework to support curriculum development activities. The framework depicted in Figure 1 is composed of four phases: Collect, Evaluate, Design, and Implement. The primary responsibility of the Collect and Evaluate phases is to define "what" a curriculum needs. The Design and Implement phases focus on "how" a new curriculum, or modifications to an existing curriculum, is implemented. The frame-

work is designed in a cyclic pattern to facilitate the continuous improvement of the curriculum. Continuous improvement is a requirement if a program expects to grow and remain viable in the ever-changing field of Information Technology. It is also a requirement for any program planning to attain accreditation (ABET/CAC, 2006).

The process model described in this paper is based on an iterative and incremental systems development model. The iterative and incremental model is an intuitive approach to the historic waterfall systems development model, similar to the spiral model proposed by Boehm (1988). Multiple development cycles take place in order to deliver a product. Cycles are divided into smaller, more easily managed iterations. Each iteration passes through each of the "standard" lifecycle phases. Subsequent iterations build on the initial product produced during the previous iteration. The product in this case happens to be an IT curriculum.

The process model described in this paper starts with the understanding and adoption of the model curriculum and proceeds through each of the four phases to deliver the product – an effective and responsive IT curriculum.
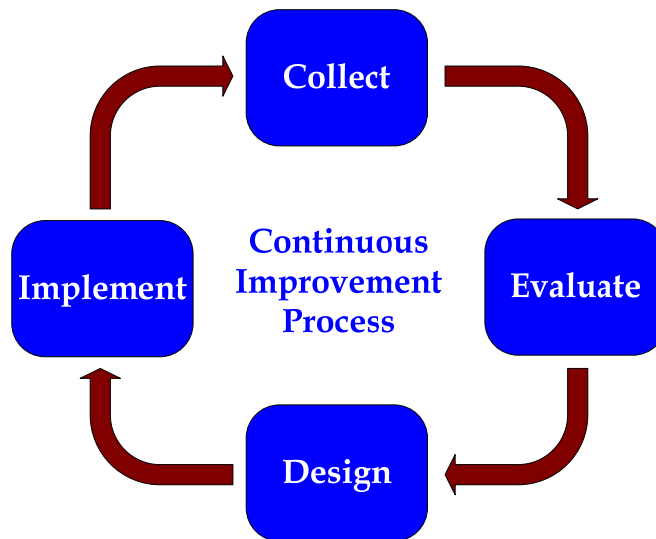

Figure 1. Continuous Improvement Process Framework

Each of the four phases, described in detail in the following sections, may be delineated by the inputs required to begin the phase, the tools and techniques used, and finally the outcomes that result. Although Figure 1 shows each phase as a distinct process without overlap, in practice there can be significant overlap. For example, it may make sense to not wait until all information is collected before beginning to evaluate the results and start formulating some design considerations.

Before proceeding through the individual phases, it is important to consider the "costs" of using this framework. The costs are described in terms of the time allocated for various activities.

In the case of CIT at Purdue, the department is undergoing accreditation for the first time in its history. The costs in this situation are notably higher than what they are likely to be during off-accreditation cycles. For example, to adequately prepare for this process, an individual faculty member was assigned full-time leadership duties as the Assessment Coordinator. Along with the Coordinator, two other faculty members formed the department's Assessment Team. The team met weekly for a semester to define all aspects of the continuous improvement and now meets about once a month to keep the process moving.

The Coordinator developed several web-based survey instruments to automatically collect and analyze data from different stakeholders. Paper-based surveys were used in situations where the web-based surveys could not be administered effectively. Several faculty dedicated some class time to administer the surveys and improve the response rates. Additional time was required for clerical staff to log the results via web tools for automatic tabulation and analysis. The web and paper survey results were used during evaluation.

All faculty had to dedicate time to support the process in addition to their normal faculty duties. The Coordinator held three half-day workshops that all faculty were required to attend to learn about the process or new supporting tools, answer questions, and get faculty buy-in. At the conclusion of each semester, each course instructor had to submit a 1"-3" binder of organized course materials that included student work for all course activities. Throughout this process, the Coordinator had to ensure that materials were being collected and organized following the departmental standard. To support all of these activities, the Coordinator developed many materials such as a quick start guide for faculty, instructions on using various web tools, instructions for administering surveys in class, reports to the advisory board, and a 300+ page self-study report.

Once the initial investment has been made, it will be important to keep an individual in place with the responsibility for overseeing the process and ensuring that tasks are done and completed on time. This person should be minimally allotted a 0.25 FTE to keep things moving. A support team of two or three additional faculty members who meet at least once per month will strengthen faculty buy-in for the long-term.

Collect Phase

Whether an educational institution is developing a new program or improving an existing program, the process begins with data collection—discovering "what" a curriculum must include. If establishing a new program, there must be a complete investigation of the market needs for the new program. Floyd (2004) describes how the University of Arkansas at Fort Smith conducted a funded research project to create a model for a competency-based IT education program. The results produced a program that equipped students with a variety of marketable skills to support both large and small employers in local, regional and national markets. Catanio (2005) discusses how LaSalle University worked with its board to analyze the needs of prospective employers who typically hired students that completed LaSalle's graduate certificate program.

Both examples illustrate the importance of a thorough requirements analysis to the subsequent design of a new program. The requirements analysis should identify stakeholders that likely include the following groups who will be impacted in some way by the new program:

- Students who enroll in the program

- Potential employers of graduates of the program

- Educators who deliver courses for the program

- Administrators who manage all aspects of program delivery

- Citizens who impact or are impacted by the program

- Accrediting bodies that evaluate and certify the program

It is important to understand that each stakeholder has a different interest in and perspective on the curriculum, so each plays a different role in defining curriculum objectives. For this reason, data must be collected from every type of stakeholder, where possible, to build a complete picture of the environment in which a program must operate. The manner by which the stakeholder data is collected for new programs will differ from how data is collected for existing programs. For

example, an agent of the school and/or community may suggest the need for the new program based on a self-initiated or published market analysis, while the program administrators will mandate regular data collection from all stakeholders of existing programs.

Each stakeholder of an existing program can make different contributions to curriculum development activities, so it is important to understand each stakeholder's perspective. Students enroll in a program for a variety of reasons, but most students have a reasonable expectation that their degree will qualify them for a career in the discipline. Employers can provide information about the jobs they expect to be available in a few years and the types of skills those positions require. Educators are both the primary authors of the curriculum and the delivery vehicle. Administrators manage the delivery of the curriculum with available resources, support the marketing of the program to potential employers, and ensure compliance with the mission and strategic plans of the university and program. Citizens can share the unique needs and expectations of the local and state community. The accrediting body is a broadly recognized, external group that gives formal recognition to a program that signifies the program's competency to deliver the curriculum. It evaluates the curriculum against predefined standards and considers much more than the curriculum before certifying a program.

Reichgelt, Lunt, Ashford, Phelps, Slazinski, and Willis (2004) assert that new programs emerge spontaneously to address perceived needs. Nonetheless, for a school embarking on the development of a new program, there are numerous curriculum models available that should be reviewed. These models are particularly helpful in differentiating an IT program, especially where similar programs already exist. Dickey (2005) compiled information about several model curricula for a variety of computing programs. Lawson and Lund (2003) differentiate computer science, management information systems, and information technology programs graphically by comparing and contrasting the amount of coverage of key topics that include business, networking, hardware, and software, among others. The ACM Joint Task Force for Computing Curricula (2005) provides the latest curricular guidelines for Computer Engineering, Computer Science, Information Systems, Information Technology, and Software Engineering programs.

Initially, reviewing available curriculum models and mapping them to the needs of an institution and its stakeholders is required to determine which model best fits the program. This task requires quantification and prioritization of the needs identified in the market analysis and critical review of the available curricular models. Anthony (2003) shares the challenges programs face in differentiating the various computing disciplines. Schools facing similar challenges should review guidelines developed by the ACM/SIGITE Curriculum Committee (2005) to help define IT programs and distinguish IT from other computing disciplines. Kamali, Liles, Winer, Jiang, and Nicolai (2005) recommend that identifying the common requirements of all stakeholders was very useful in defining what the core requirements of their proposed program should be. Al-Rawi, Lansari, and Bouslama (2005) provide a framework that tailors a model curriculum to regional and international employment needs.

Developing an IT program from the ground up will not always be necessary. For example, there may be an existing program that can provide resources for a new IT program. Hislop (2003) suggests that the curriculum from one degree might enrich the curriculum for another (new) degree through shared and improved courses. Taffe (2002) describes how Plymouth State College's Department of Computer Science offered two degrees: Computer Science and Computer Information Systems (CIS). The CIS program was designed to teach students the business side of computing, but few students enrolled in CIS. When the department noticed a need for Information Technology graduates, they converted their CIS program into an IT program.

Reviewing the various stakeholders' needs and identifying an appropriate model curriculum for emulation is not sufficient for completion of the data collection phase. It is equally important to

be aware of relevant accreditation standards. The professional societies that advocate various curricular models are in consensus regarding the importance of accreditation as a measure of quality. The models, however, do not address assessment, teaching and learning principles, and other important factors (vanVeen, Mulder, and Lemmen, 2004).

The Accreditation Board for Engineering Technology, Inc. (ABET) has a Computing Accreditation Commission (CAC) that specifies program standards and criteria for Information Technology programs, among others. For an IT program to be accredited by ABET, it must demonstrate that all elements of the program satisfy minimum, documented expectations. In addition to curriculum, ABET will review the program's objectives, outcomes, self-assessment, student support, faculty qualifications, faculty size, faculty workload, technology infrastructure, financial resources, and institutional support and facilities. Helps, Lunt, and Anthony (2005) remind educators that the purpose of the assessment process is really continuous improvement. These results also support accreditation, which certifies a program's quality.

Because the focus of this paper is on developing the curriculum, the remainder of this section describes data collection from various stakeholders regarding the curriculum only; however, if a program needs to solicit input regarding other attributes, it should use the same contact point with the stakeholder to gather this other data.

Data Sources

The CIT program at Purdue University will be undergoing its first accreditation in fall 2006. The program's Assessment Coordinator developed a diagram that describes the ongoing data collection activities from various stakeholders who contribute to the program's continuous improvement process (see Figure 2). The diagram can be used to form the core of an effective data collection process.
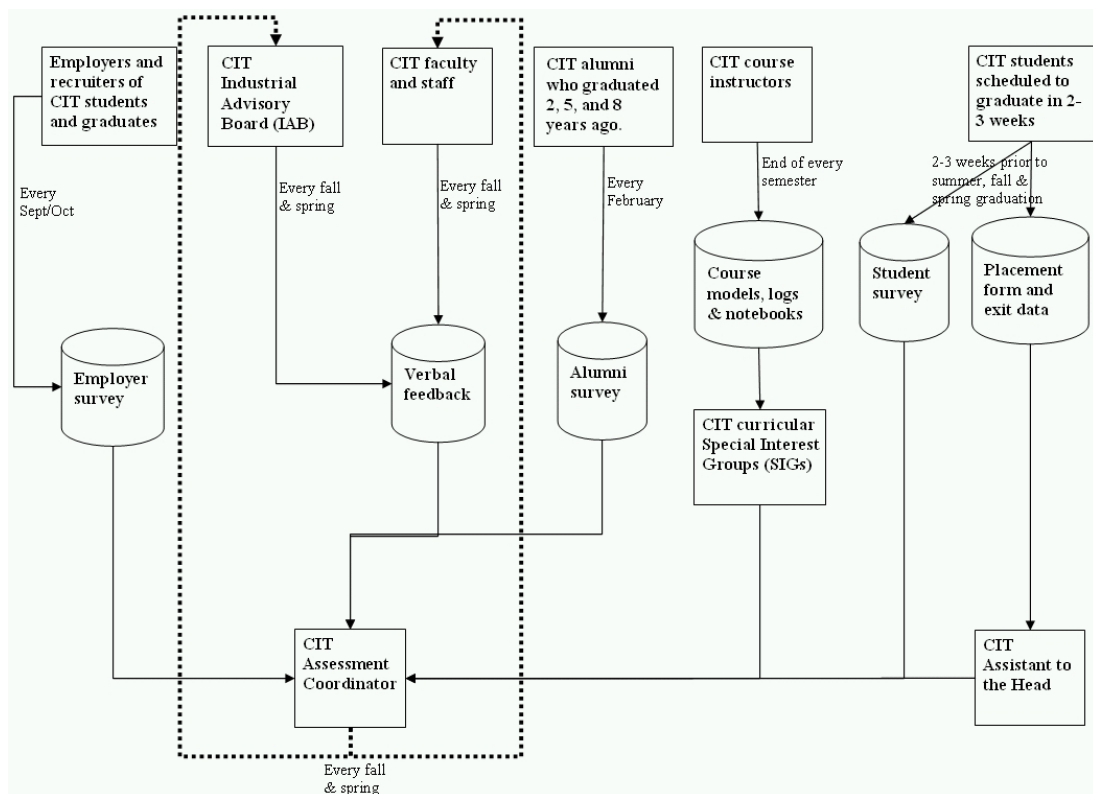

Figure 2: Data Collection from Various Stakeholders

Upon reviewing the diagram, it should be apparent that data is solicited from stakeholders in many ways. For example, students complete a placement form and an objective outcomes survey. Course instructors maintain course models and document their observations in course notebooks and improvement logs. Select groups of alumni of the program complete objective surveys. The course instructors provide verbal feedback to each other. The program's advisory board also provides verbal feedback at regular bi-annual meetings. Finally, the employers and recruiters complete an objective survey to share their opinions and observations of the Purdue CIT graduates they employ.

Although citizens and accrediting bodies are absent from the diagram, they do play an important role. The data sources shown in Figure 2 provide feedback to improve the existing program. Citizens and accrediting bodies are useful sources of benchmarking data that includes model curriculum information. Initially, this information is used to design a new program; however, in subsequent iterations of the continuous improvement process, this information can help preserve (and improve) the quality level of the program and positively impact the program's reputation.

In addition to stakeholders, a number of other data sources (as noted previously) comprise the input for designing and implementing a full curriculum. The key data sources are listed below.

- Stakeholders
  - Students
  - Employers
  - Educators
  - Administrators
  - Citizens
- Current and Emerging technologies
- Accrediting organizations
- Available resources
- Model curricula
- Community, Local and regional government needs and expectations
- Benchmarking data from other programs
- Institution mission, strategic plans

Evaluate Phase

Metrics identify the competencies against which a program will be measured. A program should document baseline metrics for all critical success factors and then identify targets. These targets may be based on the benchmarking data collected in the Collect phase or may simply be goals that the program's faculty seek to attain based upon their self-assessment of the program. Both will increase recognition by various stakeholders.

To illustrate, consider a metric that might be used to measure the national reputation of a program – the number of publications, presentations, textbooks, etc. to disseminate results of learning, engagement and discovery by faculty. The baseline measurement may show that only ten percent of the faculty engage in scholarship activities, so the program may seek to increase faculty participation to 25% within three years.

When a program is initially designed, the data collected in phase one includes model curricula, institutional mission statements and strategic plans, community needs, etc. Before the curriculum can be defined, key people should actively participate in the establishment of the program's strategic plan to ensure that it is consistent with the institution's strategic plan, matches community and regional needs, and is appropriate for the political and legal climate. The role of the model curriculum in completing this critical task is limited.

In the event an existing program is being reviewed, the same data collected from phase one can provide assurance that the program's strategic plan is still aligned with the needs of all stakeholders. This determination is made by evaluating the data against previously set standards. An important standard for most programs is to maintain a competitive environment and to align their strategies with peer educational institutions. Current competencies that were measured in the Collect phase are now assessed in the Evaluate phase for the purpose of identifying competency gaps. These gaps help determine specific targets or goals in the near and distant future that are documented in the strategic plan for the program. A subsequent iteration of the continuous improvement process allows the faculty to gauge how close the program came to attaining a goal and offers the opportunity to revise the target, as needed, based on new circumstances.

Once the strategic plan has been approved, evaluation of data for curriculum development can begin. Available course models should be evaluated for alignment with the strategic plan, availability of resources, and relevancy to the unique needs of the community and employers. For existing programs, additional tools can facilitate curricular improvements.

For example, CIT faculty use course improvement logs to document observations from their courses (see Appendix). This reflective activity is best done throughout delivery of a course, but still provides significant benefit if done as soon as the course is over. The improvement log helps future instructors of the same course identify which activities were successful and implement changes where problems were identified.

Analysis of survey data provides another valuable source of information regarding individual courses and the entire curriculum. Satisfaction surveys of employers of graduates can yield valuable knowledge regarding the program's graduates' performance against expectations. Outcomes assessment surveys of graduates can yield insights into the graduates' confidence in their abilities.

Assuming that the faculty and administrators engage in these continuous improvement assessment activities on a regular basis, the curricular evaluation can be made relatively quickly, almost in parallel with data collection. For example, CIT faculty have web-enabled many quantitative surveys, so that the evaluation of the survey data collected is immediate and automatic. Enabling the surveys for web input increases both the efficiency and the effectiveness of the process. The sooner a problem becomes apparent, the sooner appropriate actions can be taken to address the problem and implement a solution. This continuous improvement mindset results in a more effective program in the long and short run.

Design Phase

Moving to the Design phase shifts the focus from "what" the curriculum is going to accomplish (objectives and outcomes, which were defined in the Collect and Evaluate phases) to "how" the curriculum is going to be constructed or changed. The Design phase articulates the curriculum and assures that it is aligned with the mission, needs, resources and other factors identified. A program can effectively start with a model curriculum to define a core set of objectives and outcomes. Then, using the information collected and evaluated, a fully-functional, effective, tailored curriculum can be designed.

The design phase is actualized in an iterative fashion. It is overly-optimistic to believe that a complete curriculum can be designed in one cycle. The curriculum is achieved following an iterative and incremental process (see last section of the paper). Each successive cycle yields a more complete curriculum.

The ultimate design of the curriculum must support a direct association between course outcomes and program objectives. Each course model must demonstrate this association. A critical requirement is that program objectives align with the educational institution's mission and objec-

tives. If this match exists, then the course outcomes will also fit the institution's mission. This should be the case even if a direct association cannot be established, which frequently occurs due to the dynamic environment of the technology industry (Helps et al., 2005).

A program's faculty are the primary authors of the curriculum. The IT model curriculum does not specify a pedagogical approach; this is left to the faculty. It does, however, suggest four curricular approaches: practicum first, theory first, breadth first, and depth first (ACM/SIGITE Curriculum Committee, 2005). As an example, the CIT department of Purdue University has adopted a breadth first and then depth approach to curriculum design. Using this approach, students take nearly the same core IT courses during their first two years of study. The core courses provide the students with a wide range of knowledge without a great deal of depth in any one subject area. Before starting the fourth semester of the program, the students choose a major emphasis area of either network engineering technology or information systems technology. Within information systems technology, students may pursue more in-depth study in specialty areas such as data management, software development and systems integration. Each specialty area maintains its own curriculum sub-committee which reports to the department curriculum committee.

After collecting information and establishing or evolving program objectives, new course outcomes are integrated into the curriculum in appropriate existing courses, or new courses are created. A formal curriculum governance process must be established—for example a curriculum committee charged with curriculum design and implementation. The inclusion of these outcomes is reviewed by the relevant governing body and must take into consideration the program's current resources. These resources include both the instructors and the facilities. Performing a gap analysis will aid the program in determining needed resources. Because human resources and facilities are costly and can take a lengthy time to procure, the program objectives will need to be phased in over time as resources are obtained. If pursuing accreditation, part of the evaluation includes the quality and number of faculty as well as the quality of the facilities (ABET/CAC, 2006).

Because all stakeholders need to be involved in the continuous improvement curriculum process, a thorough communication plan becomes a key enabler of the Design phase. It is essential that the communication channels between faculty and other stakeholders remain open continuously during the design process. Communication is essential to facilitate a proper alignment and integration of needed learning outcomes to curriculum design. As design moves through its iterative incremental process, feedback is required from stakeholders using a variety of communication channels (voice, electronic, web portal, etc.).

It is important to communicate the results of the Evaluate phase with faculty who are the designers of the curriculum. If the faculty are not in support of the program objectives or are not aware of them, even the best process will not succeed. During the Design phase the process must facilitate this communication. Helps, Lunt, and Anthony (2005) suggest that all faculty need to be actively involved in the assessment process for the continuous improvement system to succeed. In other words, when the faculty who have to implement the process are not actively involved, then the result is inadequate. Thus, faculty input is essential to the success of the process. Not all faculty, of course, need to participate in detail for all components. But each faculty member needs to be given opportunities to contribute as appropriate and to understand and assist in formulating an effective program.

For established programs, communication with students is accomplished during the daily delivery of course lectures, assignments, and labs. When this feedback suggests that an important learning outcome is not being delivered, it creates an opportunity for the instructor to improve the design during delivery of the subject matter. These changes, whether during the current course delivery or during subsequent deliveries, must be communicated with the rest of the faculty. Also feed-

back about observed deficiencies of students in post-requisite courses provides opportunities for course improvements in subsequent offerings of the pre-requisite course.

As an example of one approach, in the CIT program faculty communication is accomplished via shared electronic folders which contain both course models and course improvement logs. Improvement logs provide an ongoing mechanism for faculty to record problems and challenges and to share them during department-level curriculum committee meetings. These curriculum meetings occur at two levels—one including the entire department and another just within the emphasis area curriculum committee. Communication with other stakeholders outside the department and possibly away from campus can be facilitated via on-campus industrial advisory board meetings and/or by providing access to curriculum documents (such as course models and improvement logs) via a web-based application.

Faculty serve as both curriculum designers and implementers. Therefore, it is imperative that all previously mentioned communication channels include all faculty, so that the total impact of changes to the curriculum can be assessed and adjusted as needed before the implementation of those changes takes place in the next phase of the process.

Implement Phase

Implement is the final phase of the continuous improvement process framework. During this phase, the inputs of new course models and course changes are used by the program's curriculum governing body to enact changes to the current curriculum. A detailed curriculum document is prepared listing the current course information and the proposed changes. From a governance perspective in the CIT department, the emphasis area curriculum sub-committee is responsible for the legitimacy of the change and a department level committee ensures that proper policies and procedures are followed. All committees operate following majority parliamentary rules. It is the responsibility of all faculty and administrators to ensure that the classroom outcomes tie back to stated program level outcomes and the institution's mission.

The CIT department's implementation process requires that all faculty members, for courses they directly supervise, prepare course models with outcomes and assessment criteria. The material is discussed and voted on by the primary emphasis curriculum committee before it is brought to the department level committee. The department level committee listens to suggestions on scheduling (when is it appropriate to offer a new course or alter an existing course) but is ultimately responsible for making sure the scheduling is correct. The department level committee is also responsible for making sure that the course can be taught using available facilities or that a plan is in place to acquire the needed space and equipment.

Once the decision is made to implement changes, timing must be considered. Adding a new course, assuming the resources exist, presents several issues. The governing committee must make sure that the additional course work does not adversely affect a student's ability to graduate in a reasonable timeframe according to the institution's guidelines. An academic institution will require a certain amount of lead time to get the new class registered and scheduled. Frequency of delivery becomes a major consideration if the new course is created as a requirement for graduation. In the CIT department all required courses are offered every session except summer to allow students flexibility and improve their ability to graduate in a reasonable timeframe.

Course content changes, on the other hand, can create scheduling issues as well, especially if the changes in one course assume resident knowledge gained from a prerequisite course that has not been added yet or which the students have not had time to complete. Before offering the post-requisite course, students must have sufficient time to acquire the new knowledge from the pre-requisite course. Deleting a course may carry similar time considerations because it could possibly hinder a student's ability to graduate in a reasonable timeframe.

Timing also becomes a factor when the program must acquire new faculty, new facilities, and/or re-train existing faculty. Each of these timing factors requires, in many cases, very long lead times. The CIT department has found it challenging to find individuals with the right combination of industry experience, effective teaching ability, and academic credentials, which can lead to long candidate searches. Re-training of faculty is best done between academic sessions and also requires time for the faculty to assimilate the new information and then prepare materials for instruction.

To enhance a program's opportunities for success during the implementation phase a quality change management process needs to be utilized. A change management process involves identifying, evaluating, and managing changes throughout the improvement process. During the Collect phase changes are identified; during the Evaluate phase changes are reviewed; and during the Design phase they are further defined in the context of the full curriculum. Throughout this process, the curriculum governing control body acts as the change control board making decisions on any and all proposed changes.

Continuous Assessment

The environment in which a curriculum is developed is not static. To be fully functional and effective a curriculum needs to adapt to change. As noted previously, a major challenge to developing an effective IT curriculum is the rapidly changing technical environment in which the curriculum must function. The IT model curriculum report recognized that "new technologies are introduced continually, and existing ones become obsolete almost as soon as they appear" (ACM/SIGITE Curriculum Committee, 2005, p. 10). Other factors that contribute to the creation and implementation of a curriculum can also change, though usually not as rapidly. The significant factors that may undergo change include program mission; community, regional, national and global needs and characteristics; industry changes; changes in faculty and technical expertise and resources; and others.

The curriculum report notes that one important characteristic of IT graduates are that they "must also be extremely adaptable" (ACM/SIGITE Curriculum Committee, 2005, p. 39). Thus, being effective in this environment implies adaptation and change. That is no less true for a particular curriculum than it is for IT graduates. To maintain quality and relevancy, programs must be dynamic—that is, they must be flexible enough, and have the mechanisms for, adapting their curricula to the environment.

A continuous improvement process is a concept that, within an educational setting, is often focused on instructional improvement and adjustments to course content. Equally important for a dynamic program is continuous assessment of the curriculum as a whole. One significant value of undergoing an accreditation process is that it focuses attention at this macro level. Continuous improvement is a never-ending process that has as its overriding goal, in this context, the evaluation of a curriculum against all the components which contribute to its design. The four-step development process framework described above can be used iteratively for the purposes of establishing such an assessment process. All the components that comprise good curriculum development decisions are evaluated repeatedly through the continuous improvement process.

After the implementation of a curriculum, the triggering mechanism for another iteration of the process is the collection of data. The primary focus of the first iteration of the Collect phase in the process model is gathering data about characteristics and needs relevant to curriculum development prior to designing and implementing the curriculum. A primary objective of the Collect phase in assessment iterations is evaluating the effectiveness of the *existing* curriculum.

One significant component of data collection during the assessment process that is not part of data collection the first time through is student performance. A recommended method for assess-

ing curriculum effectiveness is through evaluating the skills and accomplishments of graduating students relative to expected educational outcomes. Also important is collecting data about the satisfaction levels of stakeholders—for example, educators, education administrators and employers. A standardized exit exam that rates IT skills, administered over a period of time, may be effective in evaluating efforts to adapt and improve a curriculum. However, a "standardized" exam would be limited to assessing the student's core knowledge, so the evaluation of the results over time would identify just the core areas that need to be addressed. In addition, the exit exam would need to change when the core curriculum changes. Therefore, an exit exam should not be the only tool used to rate an entire curriculum.

Collecting information about the technical environment is as important in the Collect phase during the iterative process as it is the first time through. This data collection is the chief means by which evaluators can learn of changes in technology.

The Evaluate phase in assessment cycles is particularly focused on effectiveness issues and must incorporate measurement over time. Important questions need to be asked and evaluated. Is there any dissatisfaction among the stakeholders over the curriculum components? Are there expected student outcomes which are not being achieved? Are there any indications of misalignment between the established curriculum and the environment? What are the emerging technologies that could and should impact course content or perhaps the curriculum at a macro level?

The Design phase in assessment iterations becomes essentially a re-design activity, with two major emphases. The first emphasis is on improvement through correcting identified deficiencies in the curriculum—and fine tuning the existing set of proficiencies. Limited resources will affect the extent to which dissatisfaction among stakeholders can be ameliorated. The second emphasis is on adaptation to changes in the environment, particularly to changes in technology. Base technologies, such as those identified in the model curriculum, will probably continue to form the core of the curriculum. Programs may decide to differentiate and distinguish themselves through education in newer technologies that have been proven but not widely used, or even unproven emerging technologies. Significant change in the curriculum will require an extensive reconsideration of all the components that contribute to an effective curriculum.

As with the Design phase, the extent of the effort during the Implement phase in assessment iterations is dependent on the conclusions reached after data collection and evaluation. Assuming that assessment and change is part of the culture supporting the curriculum and that communication mechanisms are operational, minor adjustments can be adopted relatively easily. More extensive changes may require significant effort in areas such as acquiring or reallocating resources, and preparing faculty for new teaching assignments.

The key to having a strong iterative assessment process is having a good governance mechanism. Capable governance will ensure that the process is continuous, effective, and relevant, and that metrics are evaluated and acted upon. While a curriculum may be highly effective and successful at a particular point in time, it cannot continue so without change over time.

Stepping Through the Process

The following summarizes the major steps in the process suggested by the model. As stated previously, the process must be iterative in order to respond to continuous changes in the environment. Sub-steps may need to be modified depending on the specific program structure and circumstances in which the curriculum is being developed.

Step 1: Collect Phase

1. Identify stakeholders
2. Conduct stakeholder analysis of current curriculum and new needs
   a. Surveys
   b. Market analysis
   c. Verbal feedback
3. Model Curriculum review
4. Review Relevant Accreditation Standards
5. Update Central Repository

Step 2: Evaluate Phase

1. Define relevant metrics
2. Document current status (course models, course logs, curriculum, etc.)
3. Gap Analysis
4. Identify goals and objectives
5. Update Central Repository

Step 3: Design Phase

1. Identify needed course outcomes
   a. New course or change existing course
2. Reviewed by specialty area curriculum committee
3. Present to other specialty area curriculum committees as needed
4. Review timing considerations
5. Presentation, discussion, vote by larger faculty group
6. Update Central Repository

Step 4: Implement

1. Department Curriculum Committee
   a. Review change proposals (policies, procedures, timing)
   b. Review human and facility resource requirements
   c. Review University, College, Department strategic plans for compatibility
   d. Build formal curriculum document
2. Presentation, discussion, vote by larger faculty group
3. Send curriculum document to College level curriculum committee
4. Implement changes per schedule
5. Update Central Repository

Conclusion

The IT model curriculum published by the ACM/SIGITE Curriculum Committee (2005) is clearly a valuable aid for educational institutions that are considering the development of a new IT program. It also serves as a useful minimum standard for existing IT programs. However, the generic and static nature of the model serves only as a beginning point for curriculum building. Specifically, the model does not reflect many of the significant characteristics and considerations necessary to build a fully-functional, effective and dynamic curriculum. These characteristics and issues include adaptation to the institution's mission and goals, responsiveness to a variety of specific constituent needs, and sensitivity to the availability of teaching expertise and technical

resources. In addition, the model does not incorporate a process for continuous improvement, which is necessary to respond to a continuously changing technical environment.

For developing a fully functional effective and adaptive curriculum, the authors recommend the use of a four-step continuous improvement process framework for developing a comprehensive curriculum. Such a process is responsive to a multitude of requirements, especially to the institution's stakeholders' needs—initially and throughout the life of the program. The four phases of the process—data collection, evaluation, curriculum design, and curriculum implementation—are linked within a continuous process loop. After an initial cycle of the four phases is completed, the cycle begins anew. Only through this type of mechanism can a dynamic curriculum be sustained.

The framework has only been in use for a limited amount of time but the results reported thus far are extremely positive. CIT's Industrial Advisory Board through open discussions and surveys has confirmed the process is working. CIT also administered the student (past and present) surveys in three successive semesters and again received a very positive acclamation that the curriculum is preparing students for their first job and a career that includes life-long learning.

References

ABET/CAC. (2006). *Criteria for accrediting computing programs*. Retrieved February 16, 2006, from http://www.abet.org/Linked Documents-UPDATE/Criteria and PP/C0012006-07 CAC Criteria 02-9-06_copy(1).pdf

ACM Joint Task Force for Computing Curricula. (2005). *Computing Curricula 2005: Overview Report*. Retrieved February 16, 2006, from http://www.acm.org/education/curric_vols/CC2005_Final_Report2.pdf

ACM/SIGITE Curriculum Committee. (October 2005). *Computing Curricula: Information Technology Volume*. Retrieved January 18, 2006, from http://www.acm.org/education/curric_vols/IT_October_2005.pdf

Al-Rawi, A., Lansari, A., Bouslama, F. (2005). A holistic approach to develop IS curricula: Focusing on accreditation and IT certification. *Journal of Information Technology Education, 2, 307-327*. Retrieved January 24, 2006, from http://jite.org/documents/Vol4/v4p307-327Al-Rawi73.pdf.

Anthony, E. (October 16-18, 2003). Toward differentiating the disciplines. *Proceedings of CITC4 '03,* Lafayette, IN. 1-8. Retrieved January 18, 2006, from ACM Digital Library.

Boehm, B. (1988). A spiral model of software development and enhancement. Computer, 21, 5, May, 61 – 72.

Catanio, J.T. (October 20-22, 2005). Developing LaSalle's IT graduate certificate program partnered with industry. *Proceedings of SIGITE '05,* Newark, NJ. 19-23. Retrieved January 16, 2006, from ACM Digital Library.

Dickey, M. (February 2, 2005). *Model curricula for undergraduate programs in Computer Science and related fields*. Retrieved February 7, 2005, from http://www.cs.washington.edu/homes/dickey/curricula/.

Floyd, R.C. (October 28-30, 2004). A model for Information Technology curriculum. *Proceedings of SIGITE '04,* Salt Lake City, UT. 83-89. Retrieved January 18, 2006, from ACM Digital Library.

Helps, C.R., Lunt, B.M., Anthony, D.K. (2005). ABET accreditation with IT criteria. Proceedings of SIGITE '05, Newark, NJ. 353 – 359. Retrieved January 15, 2006, from ACM Digital Library.

Hislop, G.W. (October 16-18, 2003). Comparing undergraduate degrees in Information Technology and Information Systems. *Proceedings of CITC4 '03,* Lafayette, IN. 1-8. Retrieved January 18, 2006, from ACM Digital Library.

Kamali, R., Liles, S., Winer, C., Jiang, K., & Nicolai, B. (October 20-22, 2005). An implementation of the SIGITE model curriculum. *Proceedings of SIGITE '05,* Newark, NJ. 15-17. Retrieved January 16, 2006, from ACM Digital Library.

Lawson, E., Lund, B. M. (November 2003). IT Accreditation and Model Curriculum. *Powerpoint Presentation Slides from ISECON 2003*. Retrieved January 18, 2006, from ACM Digital Library.

Reichgelt, H., Lunt, B., Ashford, T., Phelps, A., Slazinski, E., & Willis, C. (2004). A comparison of baccalaureate programs in Information Technology with baccalaureate programs in Computer Science and Information Systems. *Journal of Information Technology Education, 3, 19-34*. Retrieved January 24, 2006, from http://jite.org/documents/Vol3/v3p019-034-098.pdf.

Taffe, W. (2002). Information technology, a degree in computing. *Journal of Computing Sciences in Colleges, 17, 3, 183-189.*

vanVeen, M., Mulder, F., & Lemmen, K. (June 28-30, 2004). What is lacking in curriculum schemes for computing/informatics? *Proceedings of ITICSE '04,* Leeds, United Kingdom. 186-190. Retrieved January 18, 2006, from ACM Digital Library.

Appendix: Sample Improvement Log

CIT 999 COURSE IMPROVEMENT INSTRUCTOR LOG

Last Updated: (date and time)

By: (instructor name)

Course Number	**CIT 999**

Course Title	Programming for the Internet

Instructions

> 1. Review the Course Model to confirm that course activities meet course objectives.
> 2. Enter data and general comments into the Course History section.
> 3. Enter or update items in the Improvement Items section:
>   a. Include the Course Model reference number for the area of improvement.
>   b. Status is "open" or "completed".

Course History (semester/instructor/location/comments)

Fall 2004 *Name of Instructor* West Lafayette

The website project continues to be the most valuable learning experience to students based upon their individual website project evaluations.

Fall 2005 *Name of Instructor* West Lafayette

The technology used to demonstrate course concepts (JSP) presents technical problems as well as provides few low-cost alternatives to clients of the website project. The software SIG approves the switch to ASP.Net using C#.Net.

Improvement Items (date/Course Model reference/status/title)

1. **3/1/2005** **5.5** Apply analytic and design methodologies **Status: Open**
    compatible with organizational settings**.**

Help client put website into production Since the technology shift from ASP.Net to JSP, there are few viable (free or low-cost) options for clients to put their websites into production. We need to find a more permanent solution for the clients, especially those who are non-profit and/or community service organizations.

1/24/2005	Attended Basics of Service Learning (SL) workshop. Presenter announced Faculty Fellow Award competition.
1/31/2005	Sent email to current SL Faculty Fellows with six specific questions regarding SL, what qualifies as a service learning project, and resources to identify potential SL projects.
2/7/2005	In the absence of a response, sent follow-up email to SL Faculty Fellows offering to meet with them instead of (or in addition to) email responses. (Name) and (name) made arrangements to meet.
2/8/2005 @ 11:30	Met with (name) and (name) to get questions answered. (Name) stated that he could provide website projects related to his own SL projects. (Name) offered to set up meeting with (name) to help identify potential community resources for future projects.
2/28/2005	Met with (name) regarding potential website clients. Need to copy sample student websites to academic server for demonstration purposes and forward links to (name).
3/3/2005	Complete proposal for 2005 Service Learning Faculty Fellow Award as possible funding source for dedicated server that could be used to house client websites to give more time to build more permanent solution for them.
2/15/2006	University support staff provided space on server for permanently hosting university service learning website. Service Learning grant used to purchase space on public server to provide temporary space to clients of website projects until a permanent solution can be found.

2. **8/21/2005 5.3** Use appropriate tools and techniques to solicit   **Status: Open**
        user requirements, and plan, analyze, design, and construct
        information systems, computer applications, and networks.

Minimize technical issues related to technology used to illustrate course concepts Since the technology shift to JSP, students easily discovered ways to get classmates' assignments and the support staff was unable to develop a secure-enough environment. Furthermore, there are few viable (free or low-cost) options for clients to put their JSP-based websites into production. Switching to another technology can solve both problems.

8/15/2005	Textbook selection and technology support for students changed to support ASP.Net with C#.Net.
1/15/2006	Web hosting service that supports technology used in the class offers free web hosting for life, so space purchased to serve as temporary host for clients as they found their own web hosting solutions.

Biographies

Jeff Brewer is an Assistant Professor of Computer & Information Technology (CIT). Prof. Brewer's teaching and scholarly interests cover systems analysis and design, computer-aided software engineering (CASE), rapid application development (RAD), agile development methodologies, and IT project management. Prior to joining Purdue, Prof. Brewer spent 19 years working in several different industry environments including specialty retail, manufacturing and most recently as the IT Director for the American Association of Nurse Anesthetists in Chicago. Most of the 19 years of experience was spent in a Project Manager role. He has developed a new undergraduate course in IT Project Management based on the Project Management Institute's Project Management Body of Knowledge. Professor Brewer is a certified Project Management Professional.

Alka Harriger joined the faculty of the Computer and Information Technology Department (CIT) in 1982 and is currently a Professor of CIT and Assistant Department Head. Prof. Harriger directs the department's accreditation activities and oversees the department's Industrial Advisory Board. Her current interests include web application development, service learning, and reducing the IT gender gap.

John Mendonca is an Associate Professor in the Computer and Information Technology Department. Prof. Mendonca has extensive industrial experience in systems development and in hardware and software management. He also has full-time teaching experience with the University of Maryland, the University of Indianapolis, and Crossroads Rehabilitation Center (where he taught programming and computer skills to adults with disabilities). He currently teaches undergraduate courses in data analysis and database design, and graduate courses in the organizational impact of information technology and information technology management.