2021 ASEE ANNUAL CONFERENCE
Virtual Meeting | July 26–29, 2021 | Pacific Daylight Time
Paper ID #33267

Continuous Improvement in Academic Computing Programs is Rarely Comprehensive

Mr. Abdullah Azzouni, Oregon State University

Abdullah Azzouni is a PhD student at Oregon State University. His research interests are Computing Education, Software Engineering, and Enterprise Software.

Dr. Jennifer Parham-Mocello, Oregon State University

I am interested in computer science education research related to K-12 curriculum, diversity in computer science, undergraduate computer science learning, and adult and teacher training.

Continuous Improvement in Academic Computing Programs is Rarely Comprehensive

Abstract

Rapid advancements in computing require academic computing departments to continually improve. According to the literature, many of those departments claim to have instituted comprehensive Continuous Improvement (CI) plans by building their own or using externally developed tools to handle the process. This paper provides an exhaustive examination of the "comprehensiveness" of those comprehensive CI plans in the current literature and whether all components of CI (360-CI) are addressed. We identify eight unique CI components, but we do not find any documentation of implementing all of them in a single program. The components identified in this paper include curriculum, faculty, course, administration, research, advising, facilities, and support staff, and from the reviewed papers, the largest number of components covered in a single comprehensive CI plan was six. To highlight the disparity of coverage of these CI components within the contexts of academic computing programs, we use a literature review to present the documented interactions among components, frequency of interactions, and sharing of data between those components. Curriculum, faculty, and course CI are discussed the most and are the most tightly integrated either by being studied together or by sharing data. In contrast, facilities, research, advising, and support staff are covered the least in the literature, and thus, weakly integrated with little discussion about the details of the required data. Therefore, we conclude that most academic computing programs do not have a comprehensive CI plan and propose additional research to further explore areas of integration among CI components.

1. Introduction

Organizations strive to establish and sustain Continuous Improvement (CI) initiatives. The concepts of Continuous Improvement and Total Quality Management came to prominence upon the publishing of W. Edward Deming's book: "Out of the Crisis" [1], in which the author detailed his 14 points for management. One of the points is Continuous Improvement (CI) defined as, "constantly and forever improving the system of production and service" [1]. In this definition, CI implies both a temporal dimension, in which organizations are improving all the time, and a spatial dimension, in which organizations are improving all of their departments, units or divisions. In order to accomplish CI, Deming proposes utilizing the Plan-Do-Check-Act (PDCA) cycle for improvement at any stage [2]. PDCA is a 4-step cycle that repeats continuously through which organizations create a plan, execute it, review the results, and finally make any corrective action before starting again.

While Deming's work was mainly directed towards business, academia took notice. The terms "Continuous Improvement" and "Total Quality Management" started to show up in higher education research papers by the late 1980's and early 1990's [3]. CI then found its way into accreditation boards' criteria. For example, the Accreditation Board for Engineering and Technology (ABET) adopted its EC2000 criteria in the late 1990's emphasizing the importance of the role of CI when accrediting a program [4]. Within academia, computing programs embraced the CI trend given the rapid advances in the field. For that matter, faculty and students created many software applications to help in their own department's or program's administrative and CI tasks [5], [6]. When faced with declining enrollment [7], CI is even more important for higher education in general and computing programs in particular.

Academic computing programs are also required to obtain local, regional, national, or international accreditation as either a standalone program or part of a bigger engineering or science department. Those accreditation systems require "proving" that a program has an established CI process. This accreditation requirement, along with the fast pace of changes within the field, should be enough for any computing program to have a well-defined CI process. However, in reality, establishing a CI process is put on the backburner until it is time for accreditation, when everybody scrambles to collect information supporting the accreditation requirement. CI in those cases, is adopted for the sake of only obtaining accreditation, rather than for actually improving a program's quality. For example, 50% of the research literature reviewed for this paper proposed or described a CI effort as a result of meeting mandated accreditation requirements. Computing programs, even if oblivious to CI prior to receiving such mandates, can use the opportunity as a springboard to establish a comprehensive CI culture [6], [8]. When a seemingly healthy person receives a health scare, they usually change their eating and exercise habits for a while. Many will cease to exercise good habits once their health improves, despite knowing that they should continue with the improved behavior. Only those who do keep the good habits continue to reap the benefits of a better health. Thus, the evaluation process of an accreditation cycle should be utilized for improvement, rather than handled as an inspection [9], [10].

Some computing programs label CI as the use of data and software tools to fix issues and problems as they come up, which is not CI given the absence of a standardized methodology to close the feedback loop. The goal of such processes is to maintain the status quo rather than to improve [11]. To help in understanding CI, this paper explores the literature for the types of comprehensive CI models proposed for academic computing programs. A comprehensive model of CI addresses all angles of a program using a 360-degree view (henceforth, 360-CI). We first present a brief background about CI in academic computing programs, and then, we synthesize and organize the findings into 4 main topics recognized in the literature. Finally, we conclude by discussing the maturity of the research in the field of continuous improvement in academic computing programs and by presenting ideas for future research opportunities.

2. Background

The term continuous improvement (CI) implies the existence of a minimum quality baseline that an organization maintains and improves upon. A minimum quality baseline can mean different things to different stakeholders. This impacts the meaning of CI and how it can be achieved and researched. The modern concepts of CI were initially developed for business processes. These processes are different in many aspects from those of academic programs. In both cases, however, CI can be analyzed and implemented through two different approaches: a system approach and a process approach. In a system approach, an organization is looked at holistically as a major system with many subsystems that interact with each other and with external entities. In a process approach, CI of organizations or entities within organizations receive information as their input, process this information in a particular way, and then produce results as output [12].

A hybrid approach is also possible in which an organization's CI utilizes a system approach while different subsystems utilize a process approach.

The hybrid approach offers more flexibility for academic programs. Through a system's approach, research of academic computing programs CI was covered in the context of improving pedagogical skills [13], optimizing course material and artifacts [14], updating curricula [15], and accreditation [16]. Given the fast-paced advancement in computing, academic computing programs need to efficiently and quickly adapt and improve through a well-defined and executed CI process. ABET's computing criteria [17] provide an opportunity to identify the major components of the CI process. These different criteria include the following: students, program educational objectives, student outcomes, continuous improvement, curriculum, faculty, facilities, and institutional support. From this list, the components of the academic program are curriculum, faculty, facilities, program administration (represented by program educational objectives and institutional support), and courses (as the main source of student outcomes). Individual students are the product [18], [19] (or the customer [6], [13]) so they benefit from CI, but they, as individuals, are not being continuously improved.

Integral to any CI effort is knowing who the customer is [6]. For academic programs (including computing), some CI researchers argue that the students are the customers [6], [20]. Other researchers discuss a larger set of customers that includes both students and employers [21]. Others have divided customers into internal customers (staff and students) and external customers (employers, community, and governments) [12]. Whichever view one forms, an academic computing program must have a clear idea of who their customers or beneficiaries are in order for their CI effort to succeed [22].

3. Methodology

We selected literature in several steps to understand the research conducted on comprehensive CI models or processes for academic computing or closely relevant programs. The word program here is defined loosely depending on how each higher education institution structures its computing education unit. It can be a dedicated computing department, it could be a computing sub-unit within a department, or it could even be a collection of departments dedicated to computing education. A 360-CI model would address all CI components of such a program. Each of these 360-CI model components is an entity within a program that could potentially conduct its own CI. With those key words in mind, a list of research papers was collected.

3.1 Paper Inclusion Criteria

For a paper to be considered, it had to either detail a CI effort at an academic program or describe an empirical study showing actual improvement in one or more aspects of the program. A list of 1767 papers was initially extracted using these criteria. The list was then reviewed in multiple passes. The first pass eliminated papers based on their title and abstract leaving 81 papers to be fully reviewed.

After reviewing these 81 papers, 36 papers were deemed relevant. During the full review, 27 more papers were added as a result of reviewing references of those candidate papers of which

10 were kept. The final list included 46 papers to be evaluated on their discussion of 360-CI efforts.

3.2 Paper Evaluation

When analyzing the selected set of papers, we identified 4 critical topics and used these as a lens for analysis. These are: components of a 360-CI model, degree of coverage of each 360-CI model component, intensity of integration among different 360-CI model components, and types of data generated and utilized by each 360-CI model component.

We constructed an initial list of 360-CI model components from the ABET criteria mentioned in the background section. This list does not eliminate the possibility of other components not covered by ABET that are still important for the overall CI of a computing program. Given these components, we determine what the degree of coverage is for each component, which is mainly the percentage of papers discussing that component in a 360-CI model context.

We determine the intensity of integration of each 360-CI model component by analyzing which components are discussed together in the same paper. If two components are discussed in more papers than other combinations of components, they have a higher intensity of integration than two CI components that are discussed in one or two papers only.

Finally, we identify the types of data generated and utilized for 360-CI models, and we determine the source component of the data and the target component improved by the data. Most of the time, the data generated by a specific component is utilized by that component itself, but there are times the data is also used to benefit other CI components.

4. Results

The final list of 46 papers covers the following 8 CI components: Curriculum, Faculty, Courses, Administration, Research, Advising, Facilities, and Support Staff. Most of these CI components were identified using ABET's criteria. Curriculum, Faculty, and Facilities directly correspond to the ABET criteria with the same name. Courses correspond to ABET's Student Outcomes. Administration and Support Staff (such as lab technicians and teaching assistants of a computing program) correspond to ABET's Institutional Support. Research and Advising are not mentioned in ABET's criteria but are addressed in the literature.

In addition to the list of eight 360-CI components, we identify three more critical areas to consider in a 360-CI comprehensive plan: 1) the coverage of the 360-CI components, 2) the integration of those components, 3) the data used and generated by the components. We outline the 360-CI components first, and then, we address how often coverage of and integration and data sharing among the components are considered in the 360-CI comprehensive plans reported in existing literature.

4.1 360-CI Model Components

We identify six components of ABET's CI criteria (Administration, Faculty, Curriculum, Courses, Facilities, and Support Staff) that are studied independently or in different combinations in the literature. We consider these components and Research and Advising, which are two components not in ABET's criteria but found in the literature, as the spokes of a wheel. Wheels with strong and well-connected spokes run farther and tolerate rough roads much better than wheels with disjointed spokes.

Curriculum & Program: Curriculum is the most commonly researched component of CI mainly through two types of quality criteria: those required for Engineering accreditation, such as mastering mathematics and physics [9], and those intrinsically desired qualities in a curriculum, such as flexibility, and non-redundancy [9]. Many papers discuss cases of how their institutes implemented CI for their own curricula and the type of tools and artifacts used [6], [23]–[25], and there is a line of research focused on the type and evaluation of metrics required for curricular CI [26].

Most curriculum CI models are an extension or variation of Deming's PDCA with an adapted set of acronyms such as Data Collection-Analysis-Review-Action [21], 5-Step-Assessment [27], Gather-Read-Analyze-Make (GRAM) [28], Plan-Deploy-Measure-Improve [29], and LEAN/EDU-FOCUS-PDCA [20]. Frank et al. presented a heuristics model that is less formal in its curriculum CI approach [30]. A more practical approach of Audit-and-Fix is also presented, where objectives of a curriculum are consolidated with course outcomes. They are then compared with accreditation requirements in order to propose and implement any necessary changes for accreditation [25]. This approach is similar to another CI model consisting of assessment (conducted in its own subprocess) and redesign subprocess [31]. One paper proposes an industrial statistical process control (SPC) approach for curriculum CI, in which the input is new students and the output is graduating students [19].

Faculty: CI of faculty covers both their pedagogical skills and disciplinary knowledge [26]. A top scientist is not always a good instructor and vice versa [32]. These two aspects are not always addressed equally, and they should not. A research institute will most probably focus on faculty's disciplinary CI, while a teaching institute will dedicate more resources towards pedagogical CI [32]. When addressing pedagogy, many papers in the literature advocate for adopting specific newer teaching methodologies as student-centered education based on active learning [20], [33], the constructivist theory of experiential learning [8], [32], or that of Kolb's learning cycle [13] and doing away with teacher-centered approaches.

One comprehensive proposal uses the GRAM model to continuously improve faculty pedagogy in their own discipline by integrating their own expertise into the institution's pedagogical goals [28]. Another proposal is for teachers to simply reflect on their experience in the class and identify areas for improvement [33]. Zahraee et al. adds more structure to this approach by asking faculty members to set their own goals and then reflect on their performance meeting those goals over the last year [6].

Three more situation-specific professional development aspects of faculty CI are also addressed: accreditation, quality management, and curriculum design [24]. Faculty's training to effectively perform and complete accreditation-related tasks and activities is relevant for those programs that actually apply for accreditation. While it might be desirable in general, improving in quality management may not be relevant for many faculty members. Similarly, faculty who are not part of a curriculum design committee may not need to learn much about the topic of curriculum design.

When discussing faculty's CI, the challenge of time must be addressed. The struggle to fit anything else into a schedule is real and usually results in faculty's resistance to any form of new CI efforts [8]. One way to overcome this challenge is to provide faculty with just-in-time training on accreditation tasks and tools [24], [34]. Embedding a pedagogical expert with the faculty is another approach to help ease the time scarcity problem [30]. Another challenge to faculty CI is how they handle the pressure to improve their teaching and the assessments used for that process (student evaluation scores vs. learning outcome achievements) [19]. Placing the responsibility of creating a strategy for faculty's CI with the individual faculty member themselves is one approach to address this challenge [33]. Another is for the department to assess faculty's needs and organize professional development activities for their staff and faculty members [26].

Student feedback is the main source of data utilized for faculty CI [20], [31] in the context of engineering education. Student feedback is typically solicited towards the end of a course as part of the course evaluation [23]. Its effectiveness has been debated extensively and proposals for other sources of data on teaching quality have been suggested and implemented [33].

With all of these complex angles to computing faculty CI and possibly out of faculty's own struggle to reduce their workload, research on software tools to help in this process is popular, and such software tools have been shown to help in the coordination and tracking of faculty's progress [35]. It is also worth mentioning that providing or using such tools can help in gaining faculty's buy-in into the CI process in general by reducing their workload [5].

Course: Since most of a student's learning is conducted in a classroom for a course, course CI is considered the nucleus of CI throughout a computing department [23], where all CI efforts eventually contribute to better course offerings. Course CI is often presented as part of a wider curriculum CI plan [24], [31], [34], [36]. Improving the curriculum and its objectives will help improve individual courses [24], [32]. From an almost opposite perspective, course CI is sometimes presented as an independent process [37] reflecting an existing perception that each instructor fixes and changes their own course in isolation [36]. The bottom-up approach is where outcomes of each individual course are gathered in order to help trim redundancy when redesigning curricula [25].

Many of the research papers advocate for changing courses from content-based to learning objectives-based [25], [27], [32] in line with recent changes in accreditation guidelines. The course CI process itself is a variation of Deming's PDCA in some cases [31], [36], [37] or a simpler input/output or SPC process that measures performance of students at graduation compared to their performance when they started [18], [19], and the main type of data to improve courses is student performance and feedback [18], [38]. This information is utilized by faculty members who are typically responsible for improving their own courses independently [21], [36].

Department & Program Administration: Computing and engineering CI literature touches on aspects that their program administration should address in their CI. Those included: benchmarking and goal alignment (intra- and inter-institutional), advancing a department's mission with institutional resources and complementary programs, staff retention, recruitment practices, policy management, industry relationship building (local and national), committee structure, & management of quality [26], [28]. When managing quality, administration needs to review whether CI tasks or processes are intended to improve quality or to merely meet accreditation requirements [39]. In terms of a defined administration CI process, annual self-assessment of core administration criteria (such as the Baldridge Criteria of core leadership, skill building, baseline assessment, strategic planning, communication, & rewards and recognition) is proposed [6]. Preparing these self-assessment reports include measurable outcomes, defined measurement methods, analyzed measurement results, recommended improvement steps, and reviewed and validated past assessment recommendations [40]. Lean methodology can also be used to continually improve administration tasks [20]. Electing to get accredited by a certain organization, such as AQIP or ABET, can also help streamline administration's CI process [39].

Research: Similar to administration, research receives little attention within the context of comprehensive CI. This should not be the case given that most academic researchers (if not all) also teach classes. Faculty members do face conflicts between their different teaching and research obligations [41]. Conflicts should be resolved, and research needs to be incorporated into the learning process, where it can enrich class material and discussions [26]. Thus, the interaction between research CI and other computing CI components must be investigated. In addition to teaching, there is an intersection of facilities and research CI [26]. Any CI of research needs to address the alignment of research objectives with that of international, national, and regional research [26]. This is not the only overlap between administration and research CI as the supervision of research quality lies somewhere between the two [26]. When discussing research CI, similar to Administration, Lean methodology is proposed [41], in addition to a general CI approach of recognizing research problems, identifying sources of problems, and implementing quality correction steps [35]. While students are thought of as the main customers and beneficiaries of the education process, research outcomes are typically of interest to the granting organization and the community at large [41].

Academic Advising: When discussing academic advising, the inadequacy of tools used to evaluate student performance and improvement needs presents a main challenge to providing proper input for any CI effort [19]. The current main input for the academic advising CI process is student evaluations and satisfaction surveys, feedback that advisors should definitely review despite its documented shortfalls [20].

Facilities: Facilities such as buildings, classrooms, and laboratories are the main locations where the education process takes place. In academic computing programs, computer labs play an especially pivotal role in leveling the playing field for students with financial needs by providing them access to the core tools they need to succeed: computers [42]. Thus, facilities CI is critical in inclusive learning, as well as in enabling computing departments to produce better students and research. These facilities being assets of the main institution and not only the property of the department adds complexity and limitations into what a computing department can do for their CI [6]. To emphasize the importance of facilities CI, Arcega explicitly includes facilities in the three "legs" of CI: personnel, facilities, and procedures [9]. Similarly, to highlight the importance of facilities, an academic department with a four-member CI team dedicates one of those members to facilities CI [6].

Support Staff: Support staff CI is mainly concerned with professional development programs, and with rewards systems [26]. CI of support staff is not commonly addressed in comprehensive CI work. Support staff includes both those with administrative tasks and those with non-administrative tasks such as: lab technicians and teaching assistants. This results in some degree of alienation of those support staff from quality processes, despite being considered a main accreditation criterion according to Arcega [9], or as part of ABET's "institutional support" [17].

4.2 360-CI Model Component Coverage

The percentage of coverage of each component varies greatly, ranging from just over 45% of papers covering Curriculum CI to about 5% of papers covering Support Staff CI as shown in Figure 1. Despite many papers advocating for a "comprehensive" CI process, the most comprehensive paper discussed 6 components, while many discussed a single component at a time despite still advocating for a comprehensive approach. Students are not included as a component since the individual students are beneficiaries or stakeholders of the CI process and not part of it.

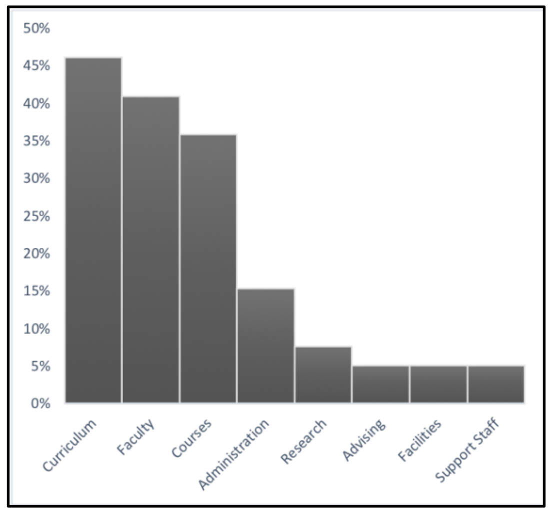

Figure 1 CI Component Coverage in the Literature

4.3 360-CI Model Component Integration

None of the models reviewed list or discuss all CI components. The largest number of CI components covered in a single paper is six [26]. Discussing more than one CI component in a single paper can highlight or uncover their dependencies, relationships, and inter-dynamics. This cannot be achieved if a CI component is missing or omitted from a paper. For this literature review, the degree to which two CI components are integrated is represented by the existence of both components in the same paper. In the set of 46 papers reviewed, the Curriculum CI component and the Faculty CI component were discussed together in 9 papers, the most of

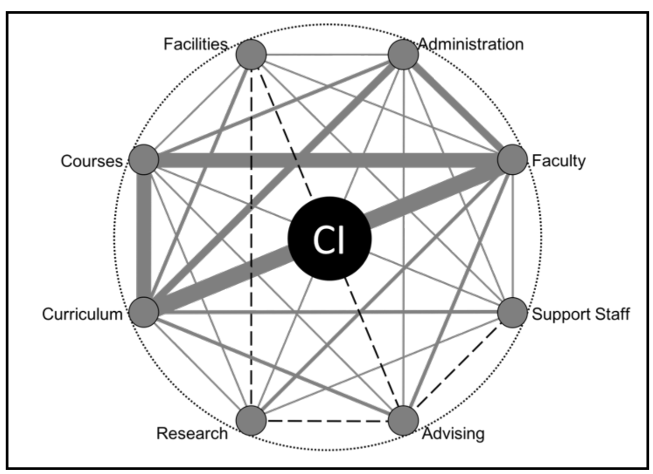

Figure 2 CI Component Integration

any two CI components. On the other hand, for example, the Research CI component and the Advising CI component were never discussed together in any of the papers. Figure 2 shows the results of how integrated a pair of CI components are. The weight of each edge represents the strength of the integration. Missing integration is represented by dashed lines. In this literature review, we find no integration between Facilities & Research, Research & Advising, Advising & Facilities, or Advising & Support Staff. As mentioned before, the strongest integration is between Faculty & Curriculum followed by the integration between Faculty & Course and Course & Curriculum.

4.4 360-CI Model Component Data

The last CI topic to discuss is the type of data each CI component utilizes and generates to help in a computing department's CI process. Throughout the literature, surveys of different stakeholders tend to be the dominant type of data. Student grades and scores are another dominant type of data utilized in CI. Figure 3 summarizes the types of data utilized for and between the different CI components within computing programs. Definitions of the different types of data are provided in Table 1. The diagram shows that little attention is paid to investigating the types of data needed for the CI of Facilities, Research, and Support Staff in the literature.

Three kinds of surveys are highlighted for Curriculum CI: senior student or exit surveys [23], [36], employer surveys [23], [31], and alumni surveys [31]. Some institutions that administer comprehensive exams for their senior students use student performance and answers to identify areas of curriculum improvement [23]. Similarly, programs that go through accreditation use the curriculum portion of the accreditation report to improve their curriculum [23]. CI of curriculum also generates data that benefit other CI components. Changes and improvements in a curriculum highlight both required and outdated faculty skills [24], [32]. A computing program can help their faculty acquire those skills or hire new faculty members who already have them. Reviewing and improving the curriculum can also alter required course objectives [13], [25], [31], [36]. Courses are potentially eliminated, or their expected outcomes are modified. There is also the potential of program or department goal modification recommendations based on the curriculum CI process [36]. A curriculum is how a computing department implements its vision and goals. Those goals can be modified by addressing issues identified in the CI process.

Faculty members are mostly trusted with their own CI. Their CI processes depend mainly on self-defined goals that faculty review at the end of the CI cycle [6]. Even without a formal step to set goals, faculty reflect on their experience in the class to evaluate where improvements are needed [28], [33]. When a more elaborate faculty CI process is implemented, faculty feedback surveys are the main form of data generated and utilized [6]. The process of faculty CI helps the administration CI by influencing hiring priorities when missing skills cannot be met internally [28].

Data that contribute to course CI can be quantitative, qualitative, or both. Student results and outcomes in a course are the main quantitative data utilized [31], [36], [38]. Students typically complete a qualitative evaluation form toward the end of the course in which they provide their feedback on different aspects of the course [6], [27].

The course teaching team also generates a whole set of qualitative data that is utilized for improving courses. Faculty's feedback about the course is the most widely used data for course CI [34]. More targeted faculty feedback is also important. This includes feedback about textbook quality and fit [13], mapping and evaluating adequacy of assessment methods to course objectives [34] and assessing student outcomes in regard to stated course objectives [31], [34].

These course qualitative evaluations help in curriculum CI. Faculty deal with teaching material and students, and the faculty are expected to provide suggestions to improve the curriculum based on this experience [36]. This can go as far as evaluating fitness of a given course to the curriculum or the program in general [23].

CI of faculty also benefits from course CI data. Students judge and sometimes grade teachers in their course evaluations [19], [23]. While these evaluations are not always accurate or objective, they do highlight areas of concerns with faculty that need improvement [23]. To mitigate the objectivity and timeliness problems, one group of researchers propose weekly student feedback forms [33].

Course feedback (by either students or faculty) also helps in the facilities' CI process. The adequacy of those facilities to meet the objectives of different courses can be evaluated, e.g., whether lab software helps students achieve course objectives [13]. Also, courses with special lab or building requirements contribute to improving facilities [23]. Even normal operating conditions of facilities (air conditioning, lighting, etc.) can be fixed and improved based on student course feedback [23].

CI of computing department administration generates data that helps in changing hiring practices [4]. The staff within the department can provide feedback through interviews and feedback forms [6]. Surveys of alumni, employers and industry also play a big part in this process [23], [36]. Community feedback is important in evaluating a department's practices, goals and mission [26]. This in turn can provide input to improving different curricula within a computing department [25], [36].

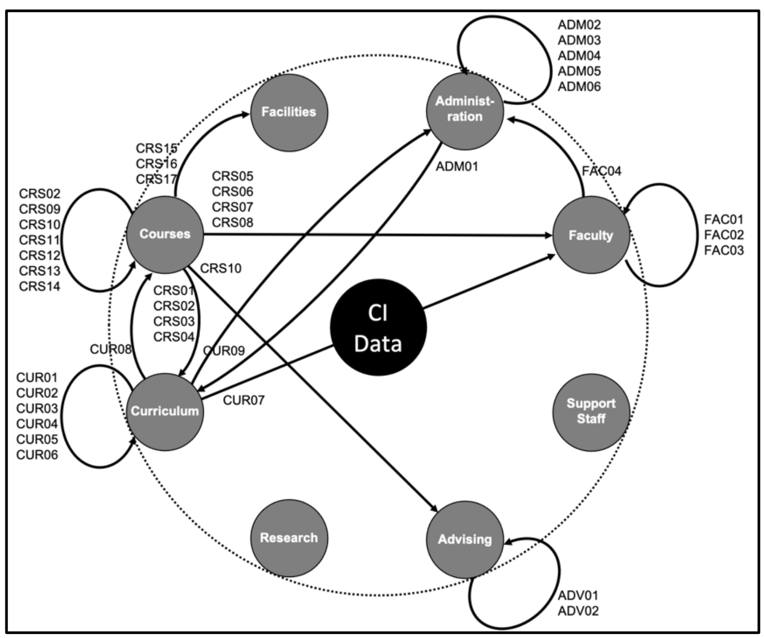

Figure 3 CI Data Types

The focus of academic advising is students. Thus, soliciting advising feedback from students frequently and towards the end of their senior year provides valuable insights into how to improve academic advising [20]. Academic advisors can also utilize student course grades and GPA to identify where students need more attention and guidance [19].

Table 1 CI Data Types

Curriculum CI Data:
- CUR01: Senior/Exit Surveys
- CUR02: Employer Surveys
- CUR03: Internship Reports
- CUR04: Alumni Surveys
- CUR05: Accreditation Report
- CUR06: Comprehensive Exams

Curriculum CI Data → Faculty CI:
- CUR07: Required Skills

Curriculum CI Data → Course CI:
- CUR08: Course Objectives

Curriculum CI Data → Administration CI:
- CUR09: Program Recommendations

Administration CI Data:
- ADM02: Hiring practices
- ADM03: Industry & Employer Surveys
- ADM04: Alumni Surveys
- ADM05: Community Feedback
- ADM06: Staff Interviews

Administration CI Data → Curriculum CI:
- ADM01: Curriculum Goals & Outcomes

Faculty CI Data:
- FAC01: Experience Reflection
- FAC02: Faculty Feedback on Own Teaching
- FAC03: Self-Appointed Goals

Faculty CI Data → Administration CI:
- FAC04: Hiring Priorities

Course CI Data:
- CRS02: Faculty Feedback
- CRS09: Textbook Quality
- CRS10: Student Results
- CRS11: Student Feedback
- CRS12: Fit of Assessment to Course Objectives
- CRS13: Student Outcomes to Course Objectives
- CRS14: Course Articulation Matrix

Course CI Data → Curriculum CI:
- CRS01: Course Adequacy
- CRS02: Faculty Feedback
- CRS03: Course Outcome Redundancy
- CRS04: Curriculum Articulation Matrix

Course CI Data → Faculty CI:
- CRS05: Student Evaluation
- CRS06: Weekly Student Feedback
- CRS07: Student Learning Outcome Achievement
- CRS08: Course Evaluation Results

Course CI Data → Facilities CI:
- CRS15: Adequacy of Facilities and Lab Software
- CRS16: Special Course Requirements
- CRS17: Course Normal Operations (A/C, lights…)

Course CI Data → Advising CI:
- CRS10: Student Results

Advising CI Data:
- ADV01 Student Evaluation
- ADV02 Senior Student Feedback & Interviews

5. Conclusion & Future Work

This paper identifies 4 critical areas to consider when applying 360-CI models in academic computing programs: 1) 360-CI components, 2) coverage of each component, 3) integration amongst these components, and 4) data necessary for each component's CI. We illustrate how most computing programs do not implement a 360-CI process that addresses the 4 critical areas identified and includes all eight 360-CI components.

Regardless of the reasons behind an educational institution's 360-CI initiative, identifying the model components helps institutions focus their resources on accomplishing the CI goals and address all 4 critical areas. Some components that might not traditionally be viewed within a computing department CI context, such as administration, whose CI can be studied from an HR or a business administration context, are included in order to understand the impact of their CI on computing programs.

The exact implementation of CI processes for each component is different because of each component's unique characteristics and stakeholders. Stakeholders can be customers in the traditional sense (such as users of a lab or classroom attendees) or can be the community or the public in general. Beneficiaries of the Research component (such as other researchers, scientists, PhD students, etc.) are different than those of the Course or Curriculum component (students, teaching faculty, etc.). Thus, designing a CI process for each component depends on that component's purpose within a computing program as well as on the mission of the computing program or institution.

Understanding how each component is covered in the literature in the context of academic computing programs helps guide future research on 360-CI. The integration among components should reduce overload and increase efficiency in accomplishing the goals of 360-CI initiatives. Optimizing how these components interact could reduce the time required for each CI, as well as reduce redundant tasks of an already busy cadre of faculty and staff. Finally, identifying data required for each component helps in facilitating access to this data by all those who need it, and optimizing the collection of data for a specific justifiable reason helps in convincing skeptics to support these processes.

An important limitation of this paper is the contrast between the sheer number of case studies and the small number of empirical research studies in this topic area. The case studies mainly describe experiences and efforts of academic institution's CI processes and initiatives that cannot be compared with each other [43]. This presents an opportunity for computing education researchers to conduct empirical research on the different CI topics identified in this paper. Further research work can validate the extracted list of components, expand the research of less researched ones, and recognize any other potential components of a computing program's 360-CI that are not included. Another important potential research area is validating the types of CI data and identifying new types of data to be utilized for CI. This can contribute to establishing a taxonomy of the types of data generated and utilized in 360-CI efforts. Scholars can use such a taxonomy to help in exploring and optimizing the dynamics and interactions of a tightly integrated set of 360-CI components.

References

[1] W. E. Deming, *Out of the crisis*. Cambridge, Mass.: Massachusetts Institute of Technology, Center for Advanced Engineering Study, 1986.

[2] R. Moen, "Foundation and History of the PDSA Cycle," presented at the Asian network for quality conference. Tokyo. https://www. deming. org/sites/default/files/pdf/2015/PDSA_History_Ron_Moen. Pdf, 2009.

[3] L. E. Coate, "TQM on campus: Implementing total quality management in a university setting.," *Bus. Off.*, vol. 24, no. 5, pp. 26–35, 1990.

[4] L. R. Lattuca, P. T. Terenzini, and J. F. Volkwein, "Engineering change : A study of the impact of EC2000," *Exec. Summery*, pp. 1–20, 2006.

[5] P. E. Maher, J. L. Kourik, and B. O. Akande, "Achieving quality, excellence, and consistency in a global academy," in *PICMET 2010 TECHNOLOGY MANAGEMENT FOR GLOBAL ECONOMIC GROWTH*, Jul. 2010, pp. 1–7.

[6] M. Zahraee, G. Neff, and S. Scachitti, “Continuous Improvement Of Engineering Technology Programs Coming Soon To A University Near You,” St. Louis, Missouri, Jun. 2000, p. 9. Accessed: Jun. 09, 2020. [Online]. Available: https://peer.asee.org/continuous-improvement-of-engineering-technology-programs-coming-soon-to-a-university-near-you
[7] K. M. Elliott, “Total Quality Management: A New Paradigm that should Enhance Marketing Education,” *Proc. 1994 Acad. Mark. Sci. AMS Annu. Conf.*, pp. 413–416, 2015, doi: 10.1007/978-3-319-13162-7_111.
[8] R. DeMichiell, R. Boisjoly, W. Ryba, and C. Tromley, “Developing And Managing Faculty-Driven Innovation: A Continuous Improvement Model With Self-Directed Work Teams,” *Educ. Innov. Econ. Bus. Adm.*, pp. 405–415, 1995, doi: 10.1007/978-94-015-8545-3_45.
[9] F. J. Arcega, “Quality and Evaluation in Higher Education,” in *Computers Supported Education*, Apr. 2016, pp. 14–24. doi: 10.1007/978-3-319-63184-4_2.
[10] P. M. Terry, “Using Total Quality Management Principles To Implement School-Based Management,” Aug. 1996, Accessed: Jun. 10, 2020. [Online]. Available: https://eric.ed.gov/?id=ED412590
[11] S. Park, S. Hironaka, P. Carver, and L. Nordstrum, *Continuous Improvement in Education. Advancing Teaching--Improving Learning. White Paper*. Carnegie Foundation for the Advancement of Teaching, 2013. Accessed: Jun. 09, 2020. [Online]. Available: https://eric.ed.gov/?id=ED560145
[12] H. J. Brits, “A model for the integration of quality management, planning and resource allocation at South African universities,” Thesis, North West University, 2010. Accessed: Jun. 07, 2020. [Online]. Available: https://repository.nwu.ac.za/handle/10394/12828
[13] R. Andersson and L. Bendix, “eXtreme teaching: A framework for continuous improvement,” *Comput. Sci. Educ.*, vol. 16, no. 3, pp. 175–184, Sep. 2006, doi: 10.1080/08993400600912335.
[14] E. García, C. Romero, S. Ventura, and C. de Castro, “Using rules discovery for the continuous improvement of e-learning courses,” in *International Conference on Intelligent Data Engineering and Automated Learning*, 2006, pp. 887–895.
[15] N. F. Veltri, H. W. Webb, A. G. Matveev, and E. G. Zapatero, “Curriculum mapping as a tool for continuous improvement of IS curriculum,” *J. Inf. Syst. Educ.*, vol. 22, no. 1, p. 31, 2011.
[16] L. Booth, J. Preston, and J. Qu, “Continuous program improvement: a project to automate record-keeping for accreditation,” in *Proceedings of the 8th ACM SIGITE conference on Information technology education*, 2007, pp. 155–160.
[17] ABET, “Criteria for Accrediting Computing Programs,” 2021 2020. https://www.abet.org/accreditation/accreditation-criteria/criteria-for-accrediting-computing-programs-2020-2021/ (accessed Jun. 07, 2020).
[18] J. M. Lombaard, “A quality model for education in engineering and technology management,” in *Innovation in Technology Management. The Key to Global Leadership. PICMET ’97*, Jul. 1997, pp. 285-. doi: 10.1109/PICMET.1997.653367.
[19] M. H. Shor and R. Robson, “A student-centered feedback control model of the educational process,” in *30th Annual Frontiers in Education Conference. Building on A Century of Progress in Engineering Education. Conference Proceedings (IEEE Cat. No.00CH37135)*, Oct. 2000, vol. 2, p. S1A/14-S1A/19 vol.2. doi: 10.1109/FIE.2000.896606.

[20] H. Shafeek, "Applying Lean Concepts in the Quality Control Systems of Engineering Education — A Case Study," in *2019 Industrial Systems Engineering Conference (ISEC)*, Jan. 2019, pp. 1–9. doi: 10.1109/IASEC.2019.8686676.
[21] Y. Kalaani and R. Haddad, "Continuous Improvement in the Assessment Process of Engineering Programs," *Proc. 2014 Am. Soc. Eng. Educ. Southeast Sect. Conf.*, p. 11, Mar. 2014.
[22] J. M. Juran, A. B. Godfrey, R. Hoogstoel, and E. Schilling, Eds., *Juran's quality handbook*, 5th ed. New York: McGraw Hill, 1999.
[23] A. Camacho and Y. Ruiz, "Facing the Challenges of Engineering Education: Continuous Improvement at Universidad del Norte," *ICEE*, 2010.
[24] L. E. Denoya, "Accreditation, curriculum model, and academic audit strategies for quality improvement in higher education," in *2005 6th International Conference on Information Technology Based Higher Education and Training*, Jul. 2005, p. F2B/7-F2B12. doi: 10.1109/ITHET.2005.1560271.
[25] J. Jawitz, N. Marais, and H. Hanrahan, "Using accreditation to drive curriculum development in engineering," *Australas. J. Eng. Educ.*, vol. 9, no. 2, pp. 137–146, 2001.
[26] I. Damaj and A. Ater Kranov, "Sustainable Practices in Technical Education: A Quality Assurance Framework," *Int. J. Eng. Educ.*, vol. 33, no. 5, Sep. 2017, Accessed: Jun. 09, 2020. [Online]. Available: https://www.researchgate.net/publication/319451363_Sustainable_Practices_in_Technical_Education_A_Quality_Assurance_Framework
[27] J. McGourty, "Strategies for developing, implementing, and institutionalizing a comprehensive assessment process for engineering education," in *FIE '98. 28th Annual Frontiers in Education Conference. Moving from "Teacher-Centered" to "Learner-Centered" Education. Conference Proceedings (Cat. No.98CH36214)*, Nov. 1998, vol. 1, pp. 117–121 vol.1. doi: 10.1109/FIE.1998.740013.
[28] J. Schreiber and L. Meloncon, "Creating a Continuous Improvement Model for Sustaining Programs in Technical and Professional Communication," *J. Tech. Writ. Commun.*, vol. 49, no. 3, pp. 252–278, Jul. 2019, doi: 10.1177/0047281618759916.
[29] I. Damaj and J. Yousafzai, "A maximum mutuality model for continuous improvement in engineering departments," in *2018 IEEE Global Engineering Education Conference (EDUCON)*, Apr. 2018, pp. 988–992. doi: 10.1109/EDUCON.2018.8363337.
[30] B. Frank, N. Simper, B. Cai, D. Salem, J. Kaupp, and M. Lindley-Peart, "Enhancing Development of Competencies by Means of Continuous Improvement Processes," *Can. Public Policy*, Nov. 2018, doi: 10.3138/cpp.2017-041.
[31] C. Zambrano, "Continuous Improvement Model to Systematize Curricular Processes in Engineering Education," in *2019 IEEE CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON)*, Nov. 2019, pp. 1–7. doi: 10.1109/CHILECON47746.2019.8988093.
[32] P. W. Hess and J. Siciliano, "A Research-based Approach to Continuous Improvement in Business Education," *Organ. Manag. J.*, vol. 4, no. 2, pp. 135–147, Sep. 2007, doi: 10.1057/omj.2007.15.
[33] J. Dinis-Carvalho and S. Fernandes, "Applying lean concepts to teaching and learning in higher education: findings from a pilot study," *Int. J. Eng. Educ.*, vol. 33, no. 3, pp. 1048–1059, 2017.

[34] A. Verma and N. Escobales, “Continuous Improvement in the Assessment and Evaluation of Engineering Technology Programs,” 2019. [Online]. Available: https://www.researchgate.net/publication/339830545_Continuous_Improvement_in_the_Assessment_and_Evaluation_of_Engineering_Technology_Programs
[35] C. S. Sitnikov, “The Management and Continuous Improvement Model of Quality in Academic Education from the Perspective of the Education-Research Process,” *ISQM 2011*, p. 135, 2011.
[36] A. Verma and G. Crossman, “An Assessment and Continuous Improvement Model for Engineering Technology Programs,” *2006 ASEE Annu. Conf. Expo. Conf. Proc.*, Jan. 2006, [Online]. Available: https://digitalcommons.odu.edu/engtech_fac_pubs/33
[37] J. Juneau and E. R. Coates Jr, “The Use of the Continuous Improvement Process in an IET Program”.
[38] T. Kunberger, K. Csavina, and R. O’Neill, “Scaffolding Assessment from Lesson Objectives through Student Outcomes: Faculty Perspectives of a Viable Continuous Improvement Model,” *J. Prof. Issues Eng. Educ. Pract.*, vol. 140, no. 4, p. A2513001, Oct. 2014, doi: 10.1061/(ASCE)EI.1943-5541.0000151.
[39] H. Methuku and H. R. Hussien, “Study on Traditional Versus Continuous Accreditation Process & Exploring Leadership Disparity,” *Int. J. Res. Commer. IT Manag.*, vol. 2, no. 1, p. 7, 2012.
[40] M. Luther and J. Wells, “Library Continuous Improvement through Collaboration on an Institution-Wide Assessment Initiative,” in *Proceedings of the 2018 Library Assessment Conference: Building Effective, Sustainable, Practical Assessment: December 5–7, 2018, Houston, TX*, Oct. 2019, pp. 48–56. doi: 10.29242/lac.2018.5.
[41] S. Nadeau, “Lean, Six Sigma and Lean Six Sigma in Higher Education: A Review of Experiences around the World,” *Am. J. Ind. Bus. Manag.*, vol. 07, no. 05, Art. no. 05, 2017, doi: 10.4236/ajibm.2017.75044.
[42] B. Moissinac, J. Parham-Mocello, and R. Pappas, “CS Student Laptop and Computer Lab Usage as a Factor of Success in Computing Education,” in *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*, Portland, OR, USA, Feb. 2020, pp. 353–359. doi: 10.1145/3328778.3366854.
[43] M. McLean, “Continuous Improvement in Higher Education: A Change Model Using Predictive Analytics to Achieve Organizational Goals,” EdD, St. John Fisher College, 2017. [Online]. Available: https://fisherpub.sjfc.edu/cgi/viewcontent.cgi?referer=https://scholar.google.com/&httpsredir=1&article=1307&context=education_etd