Data visualization and faculty engagement in program assessment

Abstract

One of the biggest challenges of program assessment is how to engage program faculty in the assessment process. The author describes how his department took advantage of the ABET Computing Accreditation Commission (CAC) Criterion 3 (Student Outcomes) changes to review their program assessment process and make changes to make the process more actionable and meaningful to the faculty. The changes include the use of performance indicators (PIs) for each student outcome, and the use of data visualization techniques to get faculty involved in evaluating the assessment data. First, the university-level and department-level program assessment processes are described. Next, the adoption of best practices in assessment, evaluation, and continuous improvement are discussed. The tools and techniques used to gain faculty buy–in are shared. It is hoped that faculty can adopt some of these best practices into their respective assessment processes.

Context: Assessment of Academic Programs and Program Review

All programs (both undergraduate and graduate) at California State University, Chico are required to submit annual assessment reports. Reports covering the previous academic year are due early in the Fall semester of the current academic year. The university provides a template for these reports.

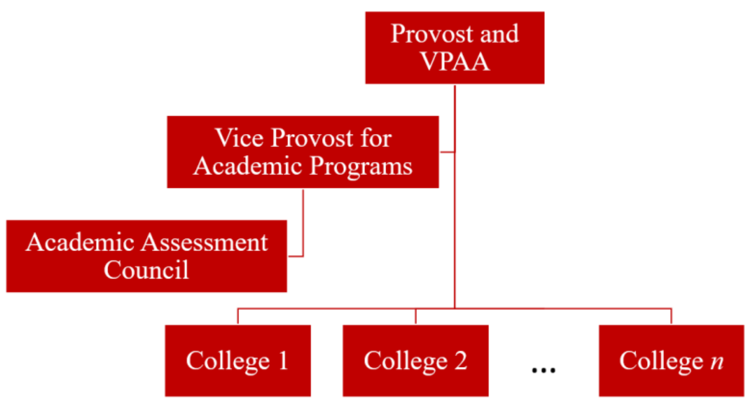


Figure 1 Organizational structure in place to facilitate program assessment.

The campus Academic Assessment Council (AAC) has been instrumental in facilitating this annual process. The AAC is a campus-level committee with representation from each of the academic colleges (typically either an associate dean or a faculty representative). The AAC chair is a faculty member. The Vice Provost for Academic Programs sits at this committee to represent the Provost's office. The program assessment reporting templates were designed by this committee.

All academic programs at our institution are subject to its program review process unless the program is accredited. Self-study reports for accredited programs count as self-study reports for program review. The program review cycle is 5 years. Results from the annual assessment reports are used in the self-study reports for program review and accreditation purposes.

Our department offers a B.S. in Computer Science (CSCI), a B.S. in Computer Information Systems (CINS), and an M.S. in Computer Science (MSCS). Our recent Fall semester enrollments have been around 400 for CSCI, 110 for CINS, and 15 for MSCS[*]. Both our CSCI and CINS programs are accredited by ABET's Computing Accreditation Commission (CAC)[1,2]. We submit program review self-study reports for our MSCS program based on our campus program review schedule.

ABET's Computing Accreditation Commission (CAC) made some changes to Criterion 3 (Student Outcomes) and Criterion 5 (Curriculum) between the 2018-19 cycle[3] and the 2019-2020 cycle[2]. Student outcomes "describe what students are expected to know and be able to do by the time of graduation. These relate to the knowledge, skills, and behaviors that students acquire as they progress through the program."[2,3] Through the 2018-2019 cycle, ABET CAC had the following student outcomes for computing programs:

Table 1 ABET CAC Criterion 3, Student Outcomes, 2018-2019

The program must enable students to attain, by the time of graduation …	
(a)	An ability to apply knowledge of computing and mathematics appropriate to the program's student outcomes and to the discipline.
(b)	An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution.
(c)	An ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs.
(d)	An ability to function effectively on teams to accomplish a common goal.
(e)	An understanding of professional, ethical, legal, security and social issues and responsibilities.
(f)	An ability to communicate effectively with a range of audiences.
(g)	An ability to analyze the local and global impact of computing on individuals, organizations and society.
(h)	Recognition of the need for and an ability to engage in continuing professional development.
(i)	An ability to use current techniques, skills, and tools necessary for computing practice.

Additionally, for the two undergraduate programs offered by our department, the ABET CAC Computer Science Program Criteria[3] includes the following student outcomes:

- **(j)** An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.
- **(k)** An ability to apply design and development principles in the construction of software systems of varying complexity.

---

[*] We did not admit graduate students temporarily for two years to work on revamping it; we just admitted our first batch of graduate students into our updated program this Fall 2019 semester.

and the ABET CAC Information Systems Program Criteria[3] includes the following student outcome:

(j) An understanding of and an ability to support the use, delivery, and management of information systems within an Information Systems environment.

Our department program assessment plan includes a curriculum map for the programs we offer. Our curriculum maps identify the courses where we regularly gather embedded assessment data from. Each semester, embedded assessment data was collected for all student outcomes from all courses that map to those outcomes. In addition to our embedded assessment data from courses identified by our curriculum maps, we use the following standardized examinations and a senior exit survey for assessment data:

Program	Standardized Examination	How we use it
CSCI	Educational Testing Service's (ETS) *Major Field Test* (MFT) in Computer Science[4]	The Comparative Data Guide for the MFT provides us with a way to map three assessment indicators to two student outcomes.
CINS	Institute for Certification of Computing Professionals (ICCP) *IS2010 Curriculum Standard Exit Exam*	Results come with a mapping of its 516 questions to each of the ABET CAC student outcomes (a) through (j)[3].

Our college office manages our Graduating Senior Survey. The survey includes questions that ask students to self-report their perceptions of the degree of attainment of student outcomes and questions that pertain to level of satisfaction for various services we offer at the department, college, and university.

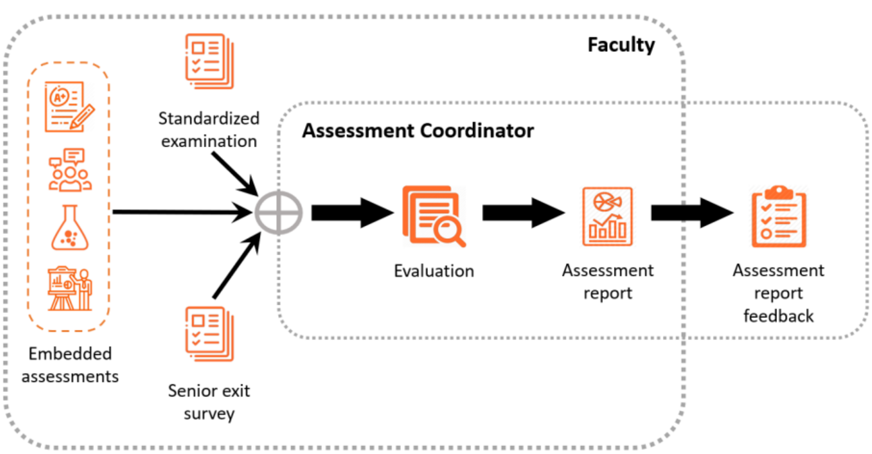


Figure 2 Annual assessment process showing data sources and responsible parties at the department level.

Figure 2 summarizes our annual assessment process and indicates our data sources and responsible parties from our department's perspective. The department's assessment coordinator, typically a faculty member who is also the chair of our department Assessment Committee, or the department chair, leads our program assessment processes. This person coordinates the gathering of embedded assessment data using a Google *Forms*[5] survey, combines this embedded data with our other data sources, analyzes the collective data gathered and presents the results to the faculty for evaluation and discussion. Any changes that need to be made to our curriculum, to our assessment schedule, or our assessment process are noted. Our Assessment Coordinator incorporates all this information into our campus-required annual assessment report for each program. These reports are reviewed by the faculty and formally approved for submission to the Office of the Vice Provost for Academic Programs (VPAP) and to our college Dean. The college's AAC representative reviews all the reports submitted from our college and completes an Assessment Report feedback rubric to identify commendations and provide constructive feedback for any areas that could be improved.

Table 2 ABET CAC Criterion 3, Student Outcomes, 2019-2020

Graduates of the program will have an ability to …
1. Analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions.
2. Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline.
3. Communicate effectively in a variety of professional contexts.
4. Recognize professional responsibilities and make informed judgments in computing practice based on legal and ethical principles.
5. Function effectively as a member or leader of a team engaged in activities appropriate to the program's discipline.

Meanwhile, the ABET CAC decided to make some improvements to their accreditation criteria. A number of 2018-2019 student outcomes were incorporated into various sections of the updated ABET CAC 2019-2020 Criterion 5, Curriculum. This resulted in a reduction in the number of Criterion 3, Student Outcomes, as listed in Table 2.

Again, for the two undergraduate programs offered by our department, the ABET CAC Computer Science Program Criteria[6] includes the following student outcome:

6. **Apply** computer science theory and software development fundamentals to produce computing-based solutions. [CS]

and the ABET CAC Information Systems Program Criteria[6] includes the following student outcome:

6. Support the delivery, use, and management of information systems within an information systems environment. [IS]

<u>Leveraging ABET Criterion 3</u>

In Fall 2018 we took the ABET CAC Criterion 3 changes as an opportunity to review our program assessment processes and consider where potential improvements can be made. The first decision our department made was to adopt the “new” ABET CAC Criterion 3 (Student Outcomes) “1 through 6” (see Table 2) for the 2018-2019 cycle, even though we could have still used the “(a) through (k)” student outcomes (see Table 1). Since this change in student outcomes was inevitable, we went ahead with the change that semester.

ABET Program Assessment workshops[7] point out that best practices in program assessment show that more meaning can be derived from the assessment of student outcomes achievement if performance indicators[8] are defined for each student outcome[9]. Performance indicators (PI), also known as key performance indicators (KPI), are specific performance measurements that pertain to the student outcome they support. PIs are measurable as degrees of attainment and PIs support a specific student outcome. The advantage of using PIs is that if the corresponding measure is below a specified threshold and the faculty agree that this is a concern, then the faculty can be specific about potential curricular adjustments they can make for students to perform better in the future. Studies show that three PIs may be sufficient for each student outcome.

To facilitate the selection of PIs, we decided to use the *ACM Computer Science Curricula 2013* (*a.k.a.* ACM CS2013)[10]. The ACM CS2013 provides a list of topics to be covered by a model computer science curriculum. It was developed with significant input from industry, educators and other stakeholders and represents best practices in terms of Computer Science education. The ACM CS2013 Body of Knowledge is organized into a set of 18 Knowledge Areas (KAs) corresponding to topical areas of study in computing. KAs are organized by Knowledge Units (KUs) with a corresponding set of Learning Outcomes (LOs). These LOs are specific and measurable enough to be considered as PIs or as least as a basis for a potential PI. For example, for the KA *Software Development Fundamentals* (SDF) there are four KUs:

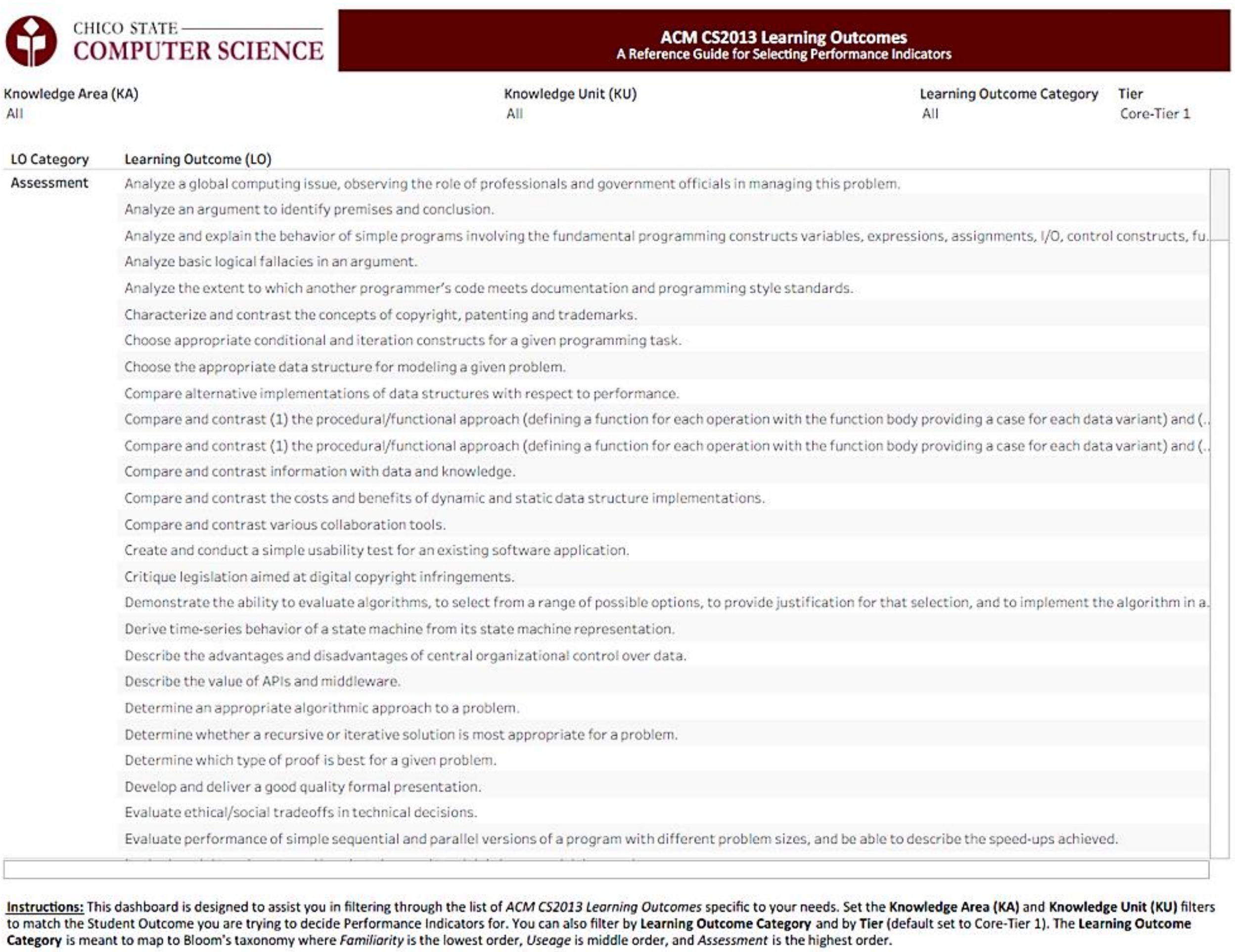
1. Algorithms and Design
2. Development Methods
3. Fundamental Data Structures
4. Fundamental Programming Concepts

Selecting the KU *Fundamental Data Structures*, we get the following LOs:

1. Analyze and explain the behavior of simple programs involving the fundamental programming constructs variables, expressions, assignments, I/O, control constructs, functions, parameter passing, and recursion. [Assessment]
2. Identify and describe uses of primitive data types. [Familiarity]
3. Write programs that use primitive data types. [Usage]
4. Modify and expand short programs that use standard conditional and iterative control structures and functions. [Usage]
5. Design, implement, test, and debug a program that uses each of the following fundamental programming constructs: basic computation, simple I/O, standard conditional and iterative structures, the definition of functions, and parameter passing. [Usage]
6. Write a program that uses file I/O to provide persistence across multiple executions. [Usage]
7. Choose appropriate conditional and iteration constructs for a given programming task. [Assessment]
8. Describe the concept of recursion and give examples of its use. [Familiarity]
9. Identify the base case and the general case of a recursively-defined problem. [Assessment]

Notice that for the most part, these LOs are measurable and could count as PIs for specific student outcomes. Also note that each LO can have one of three "levels of mastery" associated with them[10]:

- ***Familiarity***: The student understands what a concept is or what it means. This level of mastery concerns a basic awareness of a concept as opposed to expecting real facility with its application. It provides an answer to the question "What do you know about this?"
- ***Usage***: The student is able to use or apply a concept in a concrete way. Using a concept may include, for example, appropriately using a specific concept in a program, using a particular proof technique, or performing a particular analysis. It provides an answer to the question "What do you know how to do?"
- ***Assessment***: The student is able to consider a concept from multiple viewpoints and/or justify the selection of a particular approach to solve a problem. This level of mastery implies more than using a concept; it involves the ability to select an appropriate approach from understood alternatives. It provides an answer to the question "Why would you do that?"


CHICO STATE
COMPUTER SCIENCE

ACM CS2013 Learning Outcomes
A Reference Guide for Selecting Performance Indicators

Knowledge Area (KA)	Knowledge Unit (KU)	Learning Outcome Category	Tier
All	All	All	Core-Tier 1

LO Category	Learning Outcome (LO)
Assessment	Analyze a global computing issue, observing the role of professionals and government officials in managing this problem.
	Analyze an argument to identify premises and conclusion.
	Analyze and explain the behavior of simple programs involving the fundamental programming constructs variables, expressions, assignments, I/O, control constructs, fu..
	Analyze basic logical fallacies in an argument.
	Analyze the extent to which another programmer's code meets documentation and programming style standards.
	Characterize and contrast the concepts of copyright, patenting and trademarks.
	Choose appropriate conditional and iteration constructs for a given programming task.
	Choose the appropriate data structure for modeling a given problem.
	Compare alternative implementations of data structures with respect to performance.
	Compare and contrast (1) the procedural/functional approach (defining a function for each operation with the function body providing a case for each data variant) and (..
	Compare and contrast (1) the procedural/functional approach (defining a function for each operation with the function body providing a case for each data variant) and (..
	Compare and contrast information with data and knowledge.
	Compare and contrast the costs and benefits of dynamic and static data structure implementations.
	Compare and contrast various collaboration tools.
	Create and conduct a simple usability test for an existing software application.
	Critique legislation aimed at digital copyright infringements.
	Demonstrate the ability to evaluate algorithms, to select from a range of possible options, to provide justification for that selection, and to implement the algorithm in a..
	Derive time-series behavior of a state machine from its state machine representation.
	Describe the advantages and disadvantages of central organizational control over data.
	Describe the value of APIs and middleware.
	Determine an appropriate algorithmic approach to a problem.
	Determine whether a recursive or iterative solution is most appropriate for a problem.
	Determine which type of proof is best for a given problem.
	Develop and deliver a good quality formal presentation.
	Evaluate ethical/social tradeoffs in technical decisions.
	Evaluate performance of simple sequential and parallel versions of a program with different problem sizes, and be able to describe the speed-ups achieved.

__Instructions:__ This dashboard is designed to assist you in filtering through the list of *ACM CS2013 Learning Outcomes* specific to your needs. Set the **Knowledge Area (KA)** and **Knowledge Unit (KU)** filters to match the Student Outcome you are trying to decide Performance Indicators for. You can also filter by **Learning Outcome Category** and by **Tier** (default set to Core-Tier 1). The **Learning Outcome Category** is meant to map to Bloom's taxonomy where *Familiarity* is the lowest order, *Useage* is middle order, and *Assessment* is the highest order.

Figure 3 Tableau dashboard[11] created to assist in filtering the ACM 2013 learning outcomes.

These levels can be used when selecting potential PIs to make sure they match the program's intentions for level of mastery within corresponding courses in the curriculum, similar to how one would use Bloom's taxonomy[12]. Due to the large overall number of LOs to select from, I developed an interactive Tableau[13] dashboard[11] to make PI selection easier (see Figure 3). The dashboard can be used to filter the LOs by KA, KU, level of mastery (labeled as Learning Outcome Category in the interface), and Tier (whether core or elective).

As a "gentle" approach to the idea of using PIs in our assessment process, for the 2018-2019 cycle, our department's first use of PI's allowed the faculty to select the PI to use for the student outcome with which their course was going to be used as an assessment data source. The faculty were encouraged to use an ACM Learning Outcomes dashboard[11] to select an appropriate PI, with members of our Assessment Committee being available in case the faculty needed assistance or clarity.

Table 3 Staggered 3-semester assessment cycle.

STUDENT OUTCOMES		2018-2019		2019-2020		2020-2021	
		Fall 2018	Spring 2019	Fall 2019	Spring 2020	Fall 2020	Spring 2021
(1)	Analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions.	A	E	C	A	E	C
(2)	Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline.		A	E	C	A	E
(3)	Communicate effectively in a variety of professional contexts.	A	E	C	A	E	C
(4)	Recognize professional responsibilities and make informed judgments in computing practice based on legal and ethical principles.		A	E	C	A	E
(5)	Function effectively as a member or leader of a team engaged in activities appropriate to the program's discipline.	A	E	C	A	E	C
(6)	Apply computer science theory and software development fundamentals to produce computing-based solutions. **[CS]** Support the delivery, use, and management of information systems within an information systems environment. **[IS]**		A	E	C	A	E

Legend: (**A**)ssess, (**E**)valuate, (**C**)hange, if necessary

Prior to the 2018-2019 cycle, our department was collecting embedded assessment data for every ABET CAC student outcome. Data was collected from every course mapped for assessment for each of the student outcomes. This was a workload issue since we were looking at outcomes "(a) through (k)" for a total of 11 student outcomes for our CSCI program and 10 student outcomes for our CINS program – and each outcome had multiple courses from which we gathered assessment data from. We took the 2018-2019 cycle as an opportunity to try a staggered three-semester cycle where we gathered assessment data for half the student outcomes in the first semester, the other half the next semester, and a break from assessment data gathering the third semester. Each three-semester cycle followed an Assess-Evaluate-Change cycle where *Assess* meant we would gather data that semester, *Evaluate* meant we would discuss the combined results from the data that semester, and *Change* meant we would implement any changes that semester. Our assessment schedule is provided in Table 3.

As mentioned earlier, we have been using a Google *Forms*[5] survey to gather our embedded assessment data for the courses that map to the student outcomes we are collecting data from for a given semester. The form we used for the 2018-2019 cycle first asked the faculty member (1) which of the two programs and (2) which ABET CAC student outcome they were providing assessment data for. Once that has been established, the form presents seven questions and an optional eighth.

The first two questions in our Google Form allow the faculty respondent to identify the PI they were using, hopefully based on an ACM 2013 learning outcome, and the course activity, deliverable, or other artifact that they are using to gather the corresponding assessment data from. The next three questions in our Google Form ask the faculty respondent what the maximum number of points for the activity is, the actual raw scores, and the respondent's opinion about how well the PI worked in assessing our students' ability to meet the corresponding student outcome. The maximum number of points that could be earned is collected so that the raw data could be normalized for analysis. The respondents' opinion about how well the PI worked for a particular outcome was reviewed by the Assessment Committee in case recommendations needed to be made to consider a different PI for the next cycle. The last three questions in our Google Form ask the faculty respondent how well they felt the activity, deliverable, or other artifact aligned with the PI they selected, what changes they would potentially make to their data source, and any other notes or comments they may have about the process. These questions allow us to continuously improve our assessment processes for any adjustments we should be considering. We set up our Google Forms so that the form responses are dumped into a Google Sheet spreadsheet. This provides our Assessment Committee the flexibility of deciding how to analyze the data as we plan how to discuss the results with the faculty.

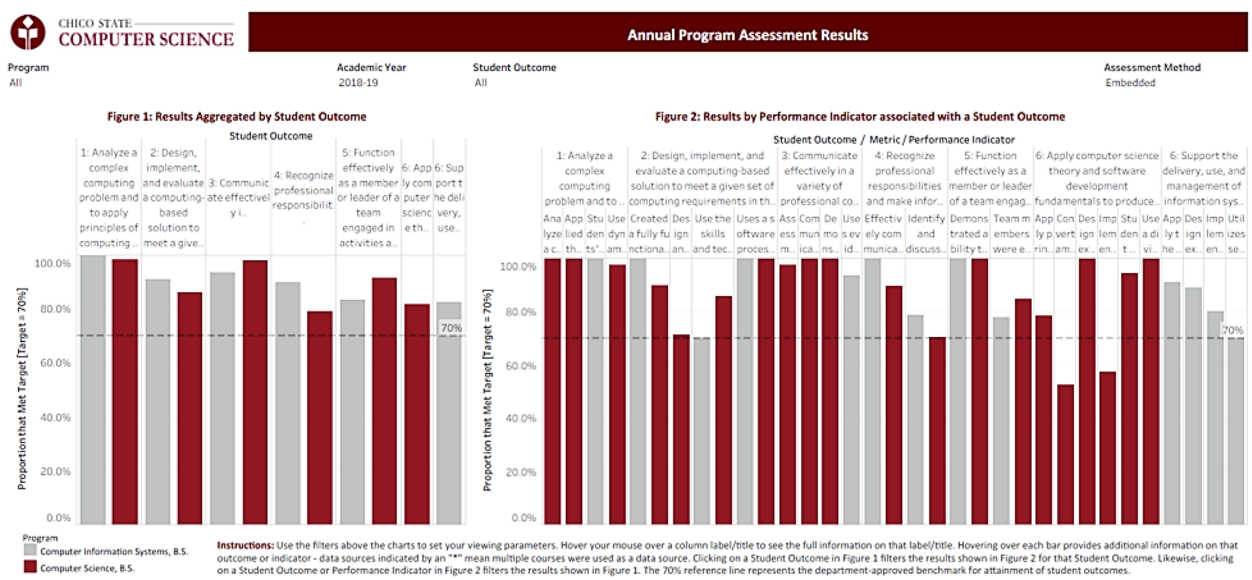


Figure 4 Assessment data dashboard shown to illustrate overall design.

Prior to the 2018-2019 cycle, our Assessment Coordinator gathered all the data and compiled them into tables and/or graphs to use in our campus annual program assessment report and for sharing with the faculty. Drafts of the reports containing these tables and/or graphs were distributed to faculty for feedback. To get faculty more engaged with our evaluation process, another change we made for the 2018-2019 cycle was to use interactive data visualization through a dashboard. Our faculty could use this dashboard to dig deeper into our results.

We use a *Tableau*[13] dashboard[14] that pulls our embedded assessment data from a Google *Sheet* and renders it the overall format shown in Figure 4. The dashboard is designed to be very interactive, taking advantage of various functionalities provided by the Tableau framework, including pop-up information that gets displayed when one hovers over various elements (*e.g.* vertical bars in the graphs) of the dashboard. Note from the top part of Figure 4 that users can set the value for four main filters that determine the results displayed by the dashboard: academic program, academic year, student outcome, and assessment method used. The filter for assessment method used has three options: (1) embedded, (2) standardized exam, and (3) survey – these correspond to our data sources depicted in Figure 2.

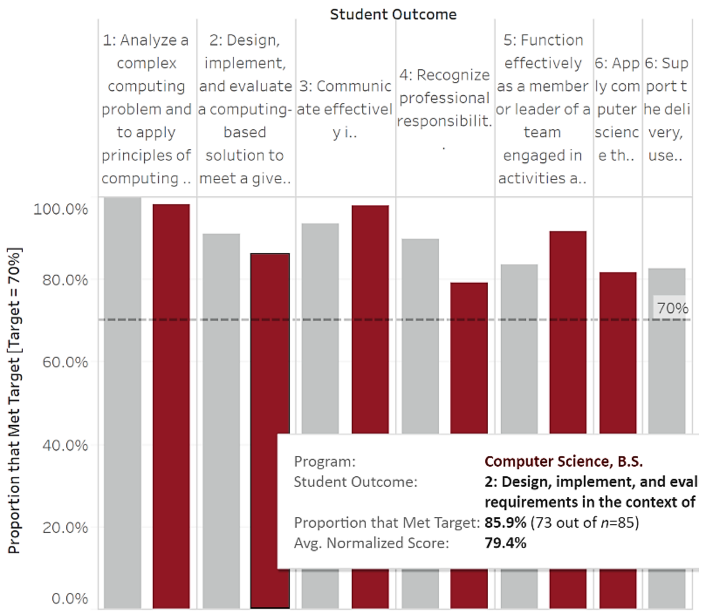


Figure 5 Assessment data dashboard - Student Outcomes pane (left pane in Figure 4).

The dashboard has two panes, labeled "Figure 1" for the left pane and "Figure 2" for the right pane. The left pane (see Figure 5) is the Student Outcomes pane that aggregates the results from all PIs for each student outcome. This pane provides one a quick overview of how our students appear to be doing overall. Our department-approved threshold of 70% is also displayed on the dashboard. Also shown in Figure 5 is an example of what pop-up information is displayed when one hovers over one of the bars in the graph (in this case, the fourth bar from the left). This on-demand information provides the user the information needed to determine how much weight to give any particular result, based on the filters set.

The right pane (see Figure 6) is the Performance Indicators pane and is best viewed when the focus (based on filter settings) is on a particular Student Outcome. Similar to the left pane, Figure 6 illustrates that the right pane has pop-up information displayed when one hovers over one of the bars in the graph. In this case, the pop-up shown is based on the fourth bar from the left where it appears the sample size is small for this PI that its low value should not carry significant weight in overall evaluation of the results. Another good use of the right pane is when the main filters are set to focus on a specific program, e.g. our CSCI program. This is illustrated in Figure 7 where for our CSCI program we could dig deeper into the two PIs that fall below the 70% department threshold.

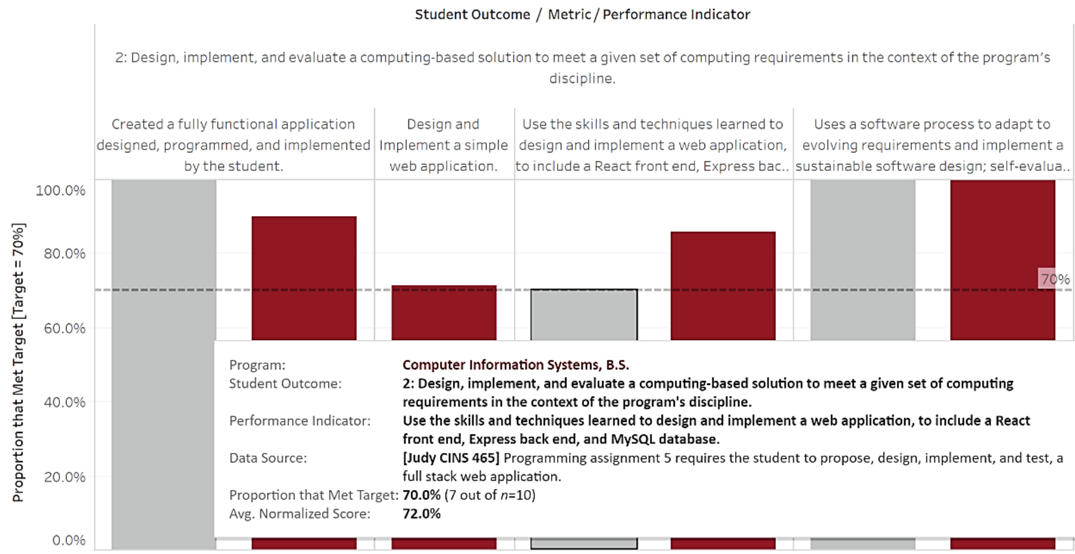


Figure 6 Assessment data dashboard - Performance Indicators pane (right pane in Figure 4) focused on a single outcome.

Once all the assessment data for our 2018-2019 cycle was collected and the dashboard set up to include this data, the faculty were informed of its availability. A demonstration of how to use the dashboard was done at a department meeting so faculty would feel comfortable drilling down into the data. Meanwhile, our Assessment Coordinator prepared the annual program assessment report, incorporating feedback from the Assessment Committee members and from faculty who used the dashboard to analyze the data. The Coordinator then presented the draft report to the faculty for discussion.

We had effective discussions about our assessment processes due to this dashboard. Examples include faculty realizing that we use data sources other than our courses for assessment – even though these have been pointed out in prior discussions of our assessment results. Other faculty better understood how all of these data sources come together to provide a better picture of student learning. We also started to question our use of the ICCP IS2010 Curriculum Standard Exit Exam for our CINS program; in particular, a number of the 516 questions we use for assessment do not apply to our curriculum. As a final example, our department decided not to use a question from our Graduating Senior Survey, as we agreed the question could be misinterpreted by students, which could skew our results. The main point here is that our faculty appeared more engaged in the assessment process because (1) they had better access to the collective data they provided and (2) they could interact with the interactive visual reports and perform their own drill downs to better understand the story behind the data.

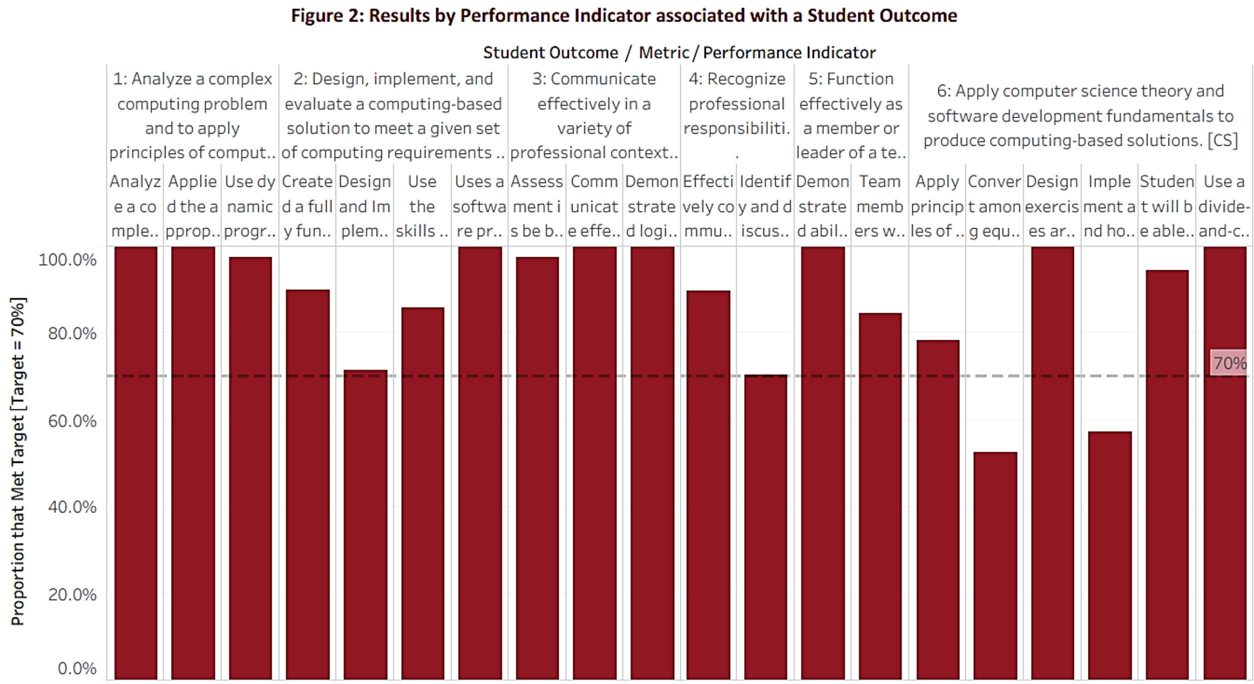


Figure 7 Assessment data dashboard - Performance indicators pane (right pane in Figure 4) focused on a single program.

Scalability

The author recognizes that other programs are housed at other institutions that have a different context than the programs presented in this paper. In this section, the author addresses the scalability of the approach presented, particularly since there are a number of other analytics and business intelligence solutions available to choose from.

Some of the *Tableau* dashboards presented in this paper actually started as Microsoft *Excel* pivot charts with sliders for filters; hence, data visualization can be provided by simply using *Excel* and these features. Dashboards can be shared by projecting the *Excel* pivot chart on a screen. If a program is housed at an institution with a Microsoft license, it is possible that the license includes access to *Power BI*[15], Microsoft's powerful business intelligence suite that can use *Excel* workbooks as data sources to generate data visualizations. With *Power BI*, dashboards can be published on the web for sharing with colleagues. A similar approach can be used by institutions that use Google (*e.g. G Suite* for Education)[16]. *Data Studio*[17] can be used to generate reports and visualizations based on assessment data stored in Google *Sheets* or even *Excel* workbooks. With *Data Studio*, dashboards ca be published on the web for sharing with colleagues.

Although our campus has a Tableau license, the dashboards presented in this paper are published through *Tableau Public*[18], Tableau's free visualization hosting platform. No license is required with this approach and the Tableau Public app, available for both Windows and Mac, is also freely available. The online hosting platform also facilitates editing of hosted visualizations via a web browser.

For the online form that faculty can use to submit their assessment data, an alternative to the use of Google *Forms* is Microsoft's *Forms*[19]. With Microsoft's *Forms* surveys are stored in *OneDrive* (or *SharePoint*) in very much the same way that Google's *Forms* surveys are stored in Google *Drive*. Microsoft *Forms* dump survey data into *Excel* workbooks for further analysis. One may also consider checking if their institution has a license for commercial survey software such as *Qualtrics*[20] before deciding to adopt a particular system for their use in assessment, particularly if there is a campus office that can assist in developing, implementing, and running these forms/surveys for you.

Next Steps

Fall 2019 provided another opportunity to reflect on our program assessment process (see our assessment schedule in Table 3). Based on our experience during the 2018-2019 cycle, we decided to undertake the following adjustments in preparation for our Spring 2020 program assessment:

1. Instead of each individual faculty selecting an ACM CS2013 learning outcome to use as a PI for the student outcome, the department agreed to vote on a set of accepted PIs for each student outcome. The intent is for the faculty to select a PI from this department-approved set. The Assessment Committee will coordinate this PI selection process to ensure sufficient coverage of the student outcome based on the PIs selected by the faculty involved.
2. The department hopes to finalize these sets of PIs for each student outcome soon so that the raw data provided by faculty is accurate and reliable. The issue in the past was that faculty provided their assessment data at the end of the semester, typically after they were done submitting grades. Hence, raw data from an artifact from an activity earlier in the semester provided little to no context – zero scores that skew the results could come from students who were absent and did not participate in the activity. It is hoped that by starting earlier and providing regular reminders throughout the semester, faculty will only collect data from students who actually did the work.

An important item to point out is that at the moment, only the current Assessment Coordinator, who happens to have some experience with data analytics and data visualization with Tableau, has been involved in the data gathering, data cleansing, and dashboard design for our assessment results. To make this a sustainable effort, particularly since the faculty appear to appreciate this level of access and involvement with the process, other faculty need to be trained or somewhat involved with this part of the process. We plan to start this training with some members of our Assessment Committee.

It is also important to point out that having access to business intelligence software like Tableau is not a requirement to get faculty involved via data visualizations (even though Tableau provides a way to create dashboards for free). Our Assessment Coordinator used to provide data visualizations from Microsoft Excel using pivot tables, pivot charts, and slicers. Such features are also now available through cloud-based solutions like Google *Sheets*.

We summarize what we have learned from this experience with the following key points:

1. Just as the program assessment process is a continuous improvement process in our efforts to ensure our curriculum is delivering the level of student learning we designed it to deliver, the assessment process itself needs to be reviewed regularly to ensure it is still providing meaningful information that faculty can act on.
2. The use of Performance Indicators (PIs) that support student outcomes is a program assessment best practice that facilitates the potential to have actionable assessment results that are impactful for our students. Student outcomes tend to still be general enough to determine where curricular adjustments need to be made. With PIs, we can more clearly identify where these potential adjustments need to be made.
3. It is important to find meaning in the assessment process you decide to adopt, but it is equally as important to settle on a process mindful of the faculty workload it requires so that it is sustainable as faculty constituency and faculty needs change. Additionally, a manageable assessment process makes it easier to ensure we have the opportunity to "close the loop" on assessment.

Bibliography

[1] ABET, "Accreditation Criteria," [Online]. Available: https://www.abet.org/accreditation/accreditation-criteria/. [Accessed January 2020].

[2] ABET, "Criteria for Accrediting Computing Programs, 2019-2020," 2 November 2018. [Online]. Available: https://www.abet.org/wp-content/uploads/2018/11/C001-19-20-CAC-Criteria-11-24-18.pdf. [Accessed January 2020].

[3] ABET, "Criteria for Accrediting Computing Programs, 2018-2019," 20 October 2017. [Online]. Available: https://www.abet.org/wp-content/uploads/2018/02/C001-18-19-CAC-Criteria-Version-1.0-12-21-17-FINAL.pdf. [Accessed January 2020].

[4] Educational Testing Service, "Major Field Tests," [Online]. Available: https://www.ets.org/mft/about. [Accessed January 2020].

[5] Google, "Forms," [Online]. Available: https://www.google.com/forms/about/. [Accessed January 2020].

[6] ABET, "Side-by-side comparison: Current criteria v. 2019-20 criteria," 18 April 2018. [Online]. Available: https://www.abet.org/wp-content/uploads/2018/12/V1vV2SideBySide_20181128.pdf. [Accessed January 2020].

[7] ABET, "Events and workshops," [Online]. Available: https://www.abet.org/events-and-workshops/. [Accessed January 2020].

[8] R. J. McBeath, Instructing and evaluating in higher education: A guidebook for planning learning outcomes, Englewood Cliffs, NJ: Educational Technology Publications, 1992.

[9] A. Driscoll and S. Wood, Outcomes-based assessment for learner-centered education: A faculty introduction, Sterling, VA: Stylus Publishing, LLC, 2007.

[10] Joint Task Force on Computing Curricula, "Computer Science Curricula 2013," Association for Computing Machinery (ACM), 20 December 2013. [Online]. Available: https://www.acm.org/binaries/content/assets/education/cs2013_web_final.pdf. [Accessed January 2020].

[11] B. Juliano, "ACM CS2013 Learning Outcomes: A reference guide for selecting performance indicators," November 2018. [Online]. Available: http://bit.ly/acmCS2013LO. [Accessed January 2020].

[12] B. Bloom, M. Engelhart, E. Furst, W. Hill and D. Krathwohl, Taxonomy of educational objectives: The classification of educational goals, Handbook 1: Cognitive domain, New York: David McKay Company, 1956.

[13] Tableau, [Online]. Available: https://www.tableau.com/. [Accessed January 2020].

[14] B. Juliano, "Computer Science Annual Program Assessment Results, 2018-2019," September 2019. [Online]. Available: http://bit.ly/csci-cins-assessment-data. [Accessed January 2020].

[15] Microsoft, "Power BI," [Online]. Available: https://powerbi.microsoft.com/en-us/. [Accessed March 2020].

[16] Google, "G Suite for Education," [Online]. Available: https://edu.google.com/. [Accessed March 2020].

[17] Google, "Data Studio," [Online]. Available: https://datastudio.withgoogle.com/. [Accessed March 2020].

[18] Tableau, "Tableau Public," [Online]. Available: https://public.tableau.com/. [Accessed March 2020].

[19] Microsoft, "Forms," [Online]. Available: https://forms.office.com/. [Accessed March 2020].

[20] Qualtrics, "Qualtrics Survey Software," [Online]. Available: https://www.qualtrics.com/. [Accessed March 2020].