GRAPH-BASED APPROACH TO MODEL THE DEPENDENCY INFORMATION OF GRADUATE ATTRIBUTES FOR SUPPORTING THE ACCREDITATION PROCESS

S. Li*, R.W. Brennan, A. Nygren
Schulich School of Engineering, University of Calgary
*Corresponding Author E-mail Address: simoli@ucalgary.ca

Abstract – In the accreditation of an engineering program, the criterion of graduate attributes is particularly challenging due to its outcome-based nature, which involves diverse instructors to collect and analyze data on students’ skills and competencies in course activities. Also, the amount of data can be vast, causing the issues of relevance and consistency of the collected data. In this context, the purpose of this paper is to facilitate the relevant process by modeling the information dependency concerning the measurements of graduate attributes and the responsibilities of stakeholders. The modeling approach is based on the graph representation that focuses on the nodes and their relations. In the graph-based model, information contents are treated as nodes, which are classified into five types: graduate attribute (GA), attribute indicator (AI), program course (PC), learning outcome (LO), and grading component (GC). Then, the contextual interpretation of the GA assessments is specified by the relations that connect these content types. In this work, three types of content relations are defined: refine, measure and associate. Further, three types of stakeholders are identified (i.e., accreditor, administrator, and instructor), along with their relations to specify their responsibilities to the content types. To demonstrate the application of the proposed graph-based model, this paper overviews the use of the Integrated Course Design Tool (ICDT) and the course outline template in the accreditation process. Based on the graph-based model, suggestions are provided toward the development of quality function deployment and software tools.

Keywords: Engineering accreditation : graduate attribute assessment : graph-based modeling : information dependency.

1. INTRODUCTION

Graduate attributes have become an important outcome-based criterion to assess an engineering program in the accreditation process [6]. By the outcome-based notion, the program administrators need to show that the graduates of their programs have acquired adequate levels of skills and competencies in view of twelve graduate attributes. In practice, it is not easy to manage the information related to graduate attributes for the following reasons.

- Participation of multiple stakeholders. As the graduate attributes criterion requires the assessment of the students’ outcomes, it heavily relied on the grading information provided by the course instructors, who may have different interpretations of graduate attributes with various levels of engagement in the accreditation process.

- Complexity of the information. There are twelve graduate attributes, each of which generally requires a minimum of three samples of assessments. Combining the contextual information of learning activities (e.g., learning outcomes and grading methods), the total amount of information is vast and complex. For instance, there could exist some logical links among different pieces of information, causing the concern of information consistency.

- Continual data collection and analysis. To support the continual improvement of the engineering programs, it is expected to collect and analyze the outcome data of graduate attributes annually. Without standardized processes, these annual routines can incur considerable administrative loads and the efforts from instructors.

- Interpretation of the graduate attribute data. While it is common to use students’ grades of their course works to quantify the assessment of graduate attributes, such quantified data may not be self-explanatory in view of graduate attributes, causing some ambiguity in the interpretation of the data.

To address the above challenges, we have implemented two closely related tools as the ongoing practice. The first one is the course outline template (COT) that standardizes the contents of a course outline. Traditionally, course outlines have been used as the formal documents to align the expectations between students and instructors, e.g., how the course works are evaluated in view of grading components (e.g., examination, project) and how these
grading components are associated with learning outcomes. In addition to this traditional use, the COT supports the accreditation process by asking the instructors to link learning outcomes to graduate attributes. By using learning outcomes as the middle connection, graduate attributes can be assessed quantitatively based on grading components in a course.

The second tool to support the accreditation process is the Integrated Course Design Tool (ICDT) [4]. In brief, the ICDT is an Excel-based software tool, which analyzes the data from a grading sheet and yields the quantitative measures of graduate attributes in the five-bin distribution (i.e., unsatisfactory, below expectation, meets expectation, exceeds expectation and excellent). This is achieved by tracking the breakdowns of grading components that are mapped to graduate attributes as regulated in the COT. In practice, the ICDT has been used to help prepare the annual reports on the evaluation of graduate attributes.

While the COT and the ICDT are helpful in the practical operations, it is also appropriate to review the roles of the COT and the ICDT for better synergy and integration in the information management. In particular, this paper will address the dependency information that makes the data collection and analysis activities difficult to track and manage. In the methodical approach, this paper applies the graph-based modeling using simple nodes and edges as the generic representation of the dependency information. In this graph-based model, each node represents an item of information contents (e.g., a learning outcome). Then, the edges are used to show the dependency information in view of the relations between two types of information contents, between contents and stakeholders, and between two types of stakeholders.

In practice, the dependency information can be often found in the accreditation process. For example, the curriculum maps required by Canadian Engineering Accreditation Board (CEAB) (i.e., Criterion 3.1.2) ask for “the relationship between learning activities for each of the attributes and semesters in which these take place” [6, p. 14]. In a graph-based model, this phrase can be visualized by representing learning activities, graduate attributes and semesters as graph nodes and using edges to connect these nodes. Such a graph can be compactly represented via a matrix (or table) format, which is a common format in the deployment function and software tools to support the ICDT and propose the future development of quality management in education such as the quality audit tool [3], quality function deployment [14], total quality management [21], and the graph theoretic model [12].

Along this type of efforts, this paper is intended to develop the graph-based model to manage the dependency information involved in the accreditation process.

The discussion of engineering accreditation can also be associated with the field of quality assurance and six-sigma management in education [11, 15]. In this context, some quality-based tools have been proposed to support quality management in education such as the quality audit tool [3], quality function deployment [14], total quality management [21], and the graph theoretic model [12].

Along this type of efforts, this paper is intended to develop the graph-based model to manage the dependency information involved in the accreditation process.

In the next section, we will first deploy the graph-based model of the information contents with an example to illustrate the notations and the dependency information. Then, Section 3 will further extend the graph-based content model with the relations of stakeholders. Section 4 will use the graph-based model to review the existing COT and ICDT and propose the future development of quality deployment function and software tools to support the activities of accreditation and continual improvements. Section 5 will provide the summary and the future work.

2. GRAPH-BASED MODEL OF THE INFORMATION CONTENTS

In mathematics, a graph (denoted as $G$) is an ordered pair such that $G = (V, E)$, where $V$ stands for a set of vertices or nodes, and $E$ stands for a set of edges, where each edge connects two nodes. To develop the graph-based model, we first itemize and categorize the information contents associated with the graduate attributes and label them as different types of nodes (i.e., $V$). Then, we classify
the edges or relations (i.e., E) among different types of nodes as a way to indicate how one type of contents is related to another type. At the end of this section, we will illustrate the graph-based model with an example.

2.1. Itemization and Categorization of Information Contents

The intent of classifying the information contents is to regulate the assessment of graduate attributes in a specified context. In this work, we classify five types of contents, which are denoted and described as follows.

Graduate attributes

CEAB has defined 12 graduate attributes, which engineering students need to demonstrate at the time when they complete the programs. Graduate attributes have covered various aspects such as scientific knowledge and skills (e.g., knowledge base, problem analysis and investigation), engineering skills (e.g., design and use of engineering tools), and professional skills and expectations (e.g., communication, ethics, and lifelong learning). To develop the notation, let $GA$ be the set of graduate attributes, each of which is denoted as $ga_i$. As CEAB has specified 12 graduate attributes, we can denote $GA = \{ga_1, ga_2, \ldots, ga_9, ga_{10}, ga_{11}, ga_{12}\}$.

Attribute indicators

On the CEAB accreditation document [6, p. 15], it is required to define “indicators” to make graduate attributes more “measurable”. We interpret this as an observable way to specify the notions of graduate attributes in the context, which allows the accreditors to interpret the collected data more meaningfully. Notably, it seems that CEAB leaves the detailed definitions of attribute indicators to the program administrators. For example, the CDIO syllabus (CDIO stands for Conceive-Design-Implement-Operate) has specified various levels of details on the knowledge and skills expected from engineering students for attribute indicators [7, 9].

To further explain, consider the graduate attribute #4: Design (i.e., $ga_4$). To assess the students’ competency in this attribute, the attribute indicators may include the following.

- Indicator #1: Define the scope of a design problem based on the limited resources.
- Indicator #2: Propose multiple solutions to an open-ended design problem.
- Indicator #3: Justify the design’s merits based on observable evidence.

Then, under the same graduate attribute Design, these indicators can differentiate what students are actually good at (or where students need improvements). For example, students can be good at brainstorming different solutions for the same problem (i.e., Indicator #2), but they may have difficulty to define the scope of a vaguely defined problem on their own (i.e., Indicator #1).

Based on the above discussion, attribute indicators are defined in the context of each graduate attribute. For the notation, let $AI_i$ be the set of attribute indicators of the $i$th graduate attribute, and let $ai_{ij}$ denote the corresponding $j$th attribute indicator, i.e., $AI_i = \{ai_{i1}, ai_{i2}, \ldots, ai_{ip} \}$. Let $AI$ be the superset of $AI_1$ and $AI = \{AI_1, AI_2, \ldots, AI_9, AI_{10}, AI_{11}, AI_{12}\}$. Notably, each attribute indicator is unique for each graduate attribute.

Program courses

Program courses are referred to the credited courses toward the completion of an engineering program. The offering and administration of program courses should be the most observable activities in the accreditation process, and CEAB has established a procedure to examine the program courses on the criterion of “Curriculum content and quality” [6, pp. 18-22]. Suppose that a program has $m$ number of courses. For the notation, let $PC$ be the set of $m$ program courses, each of which is denoted as $pc_i$, i.e., $PC = \{pc_1, pc_2, \ldots, pc_i, \ldots, pc_m\}$.

Learning outcomes

While the course descriptions are officially used to regulate a course’s materials (i.e., published on the academic calendar), it has been a common practice to articulate the learning outcomes that specify what we expect students to learn in a course. While learning outcomes may not be regulated precisely in the accreditation process, the significance of high-quality learning outcomes has been recognized as an important means for effective teaching and learning [1, 22]. Thus, it is common to require course instructors to list the learning outcomes in their courses.

For the notation, let $LO_k$ be the set of learning outcomes of the $k$th program course, and let $lo_{kp}$ denote the corresponding $p$th learning outcome, i.e., $LO_k = \{lo_{k1}, lo_{k2}, \ldots, lo_{kp}, \ldots\}$. Let $LO$ be the superset of $LO_1$, and $LO = \{LO_1, LO_2, \ldots, LO_i, \ldots, LO_m\}$. Notably, each learning outcome is unique for each program course.

Grading components

Grading components define how the performance of a student is assessed in a course such as examinations, presentations and projects. To maintain the transparency of the grading process, grading components are often specified in a course outline, and they cannot be altered after the course starts. For the accreditation purpose, it is common to use the data from grading components to assess the graduate attributes in the program.

For the notation, let $GC_k$ be the set of grading components of the $k$th program course, and let $gc_{kp}$ denote the $q$th grading component, i.e., $GC_k = \{gc_{k1}, gc_{k2}, \ldots, gc_{kp}, \ldots\}$. Let $GC$ be the superset of $GC_1$, and $GC = \{GC_1, GC_2, \ldots, GC_i, \ldots, GC_m\}$. Notably, each grading component is unique for each program course.

We consider that these five types of contents are fundamental to describe the context of the assessment of graduate attributes. Further, the accreditation requirement and the administrative coordination define the top-down control over the contents of $GA$, $AI$ and $PC$. In contrast, course instructors represent the bottom-up effort who
contribute to LO and GC. Then, it becomes important to 
rationalyze how the quantitative factors GC can be 
interpreted in the context of GA. The next sub-section will 
discuss such kinds of relations.

2.2. Relations of the Contents

In this work, three types of relations are defined to relate 
these five types of contents, and these relations are (1) 
refine, (2) measure and (3) associate. The use of “refine” 
relations is to further specify the contextual meaning of a 
content item. Typically, the “refine” relation is conveyed 
in a two-tier tree structure, where the bottom-tier nodes 
refine the content item at the top tier. Let \( R_{\text{refine}}(a, b) \) denote 
the relation that ‘b’ refines ‘a’. In this work, the “refine” 
relations are found in the following.

- **Attribute indicator, AI**, refines graduate attribute, GA, 
i.e., \( R_{\text{refine}}(ga_i, ai_i) \). This “refine” relation represents 
the effort to specify observable / measurable indicators 
for each graduate attribute.

- **Learning outcome, LO_k**, refines program course, PC, 
i.e., \( R_{\text{refine}}(pc_k, lo_k) \). This “refine” relation represents 
the effort to specify the skills and competencies to be 
learned in a course.

As a general observation, both “refine” relations are often 
required by a manager to make sense of some contents 
developed by an implementer in quality assurance. For 
example, \( R_{\text{refine}}(ga_i, ai_i) \) can help CEAB to interpret a 
quantified measure associated with a graduate attribute. 
Similarly, \( R_{\text{refine}}(pc_k, lo_k) \) can help the school administrator 
to understand the skills and competencies that an instructor 
is trying to teach in a course. In this sense, the “refine” 
relations are important to establish the common ground 
between managers and implementers.

The use of “measure” relations is to quantify a content 
item in a context. In this work, grading components are the 
only content type with quantified measures, and they are 
used to measure how well students perform per learning 
outcomes in a program course. Let \( R_{\text{measure}}(a, b) \) denote 
the relation that ‘b’ measures ‘a’. Then, one “measure” 
relation is defined in the following.

- **Grading component, GC_k**, measures learning outcomes, 
LO_k, i.e., \( R_{\text{measure}}(lo_k, gc_k) \). This “measure” relation 
represents the effort to quantify the performance of a 
student based on their skills and competencies in view 
of learning outcomes demonstrated in a course.

The use of “associate” relations is to map the content items 
between two sets. Let \( R_{\text{associate}}(a, b) \) denote the relation that 
‘a’ is associated with ‘b’. To bridge course activities to 
graduate attributes, this work specifies the mapping 
between learning outcomes and graduate attributes. Then, 
one “associate” relation is defined in the following.

- **Learning outcome, LO_i**, associates graduate attribute, 
GA, i.e., \( R_{\text{associate}}(ga_i, lo_i) \). This “associate” relation 
represents the effort to ensure that the course activities 
are contributive to the training of engineering students 
in view of graduate attributes.

Notably, matrix can be used to capture the structure of both 
“measure” and “associate” relations discussed above, 
where each matrix entry represents a link between two 
elements from two different sets. Figure 1 shows the 
graphical representation, where the nodes represent the 
contents, and the edges represent the relations.

![Fig. 1. The graph-based content model.](image)

2.3. Illustrative Example

To illustrate the use of the graph model, let us consider 
an engineering course about mechanical systems in 
buildings. First, let us label this course as pc_5. In this 
course, there are five learning outcomes, labelled and listed 
as follows.

- **lo_5,1**: Determine the heating / cooling loads and 
ventilation requirements
- **lo_5,2**: Analyze the thermodynamic, fluid mechanics and 
control properties of HVAC systems and equipment
- **lo_5,3**: Design a HVAC system for a small building
- **lo_5,4**: Recognize the codes, regulations and standards in 
HVAC design
- **lo_5,5**: Communicate the HVAC design in written and 
oral forms

Also, this course has four grading components, labelled 
and listed as follows.

- **gc_5,1**: Team project
- **gc_5,2**: Quizzes
- **gc_5,3**: Site visit
- **gc_5,4**: Final examination

In the assessment, this course is associated with three 
graduate attributes: ga_2 – problem analysis, ga_4 – design, 
and ga_7 – communication skills. Relevant attribute 
indicators are then labelled and listed as follows.

- **Attribute indicators of ga_2 – problem analysis**
  - **ai_2,1**: Apply engineering knowledge and skills to 
solve real world problems
  - **ai_2,2**: Make assumptions that successfully simplify a 
complex problem
  - **ai_2,3**: Synthesize problem solutions and formulate 
summary recommendations

- **Attribute indicators of ga_4 – design**
  - **ai_4,1**: Interpret ethical, social, environmental, legal 
and regulatory influences
  - **ai_4,2**: Select concepts and analyze the trade-offs 
among and recombination of alternative concepts

CEE18; Paper 060
University of British Columbia; June 3 – 6, 2018 — 4 of 10 —
In this work, stakeholders can relate the contents of the graph-based model in two ways: (1) define and (2) develop. By the relation “define”, stakeholder takes the managerial role to define the scope and guidance of the contents. Let \( R_{\text{define}}(a, b) \) denote that stakeholder ‘a’ defines content ‘b’. In our context, accreditors define the scope of the graduate attributes, and administrators define the set of the courses toward the completion of an engineering degree. These can be denoted as follows.

- \( R_{\text{define}}(\text{ACC, GA}) \)
- \( R_{\text{define}}(\text{ADMIN, PC}) \)

By the relation “develop”, stakeholders are responsible for developing the details of the contents and the content relations. Let \( R_{\text{develop}}(a, b) \) denote that the stakeholder ‘a’ develops the content or relation ‘b’. In our context, administrators develop attribute indicators for each graduate attribute. Also, instructors develop learning outcomes for each program course and grading components to measure learning outcomes. In addition, instructors are also required to develop the relations of learning outcomes with grading components and graduate attributes, respectively. These can be denoted as follows.

- \( R_{\text{develop}}(\text{ADMIN, AI}) \)
- \( R_{\text{develop}}(\text{ADMIN, } R_{\text{define}}(\text{GA}_i, \text{AI}_j)) \)
- \( R_{\text{develop}}(\text{INST, LO}) \)
- \( R_{\text{develop}}(\text{INST, GC}) \)
- \( R_{\text{develop}}(\text{INST, } R_{\text{measure}}(\text{LO}_k, \text{GC}_l)) \)
- \( R_{\text{develop}}(\text{INST, } R_{\text{associate}}(\text{GA}_i, \text{LO}_k)) \)

Figure 3 shows how the stakeholders are related to the graph-based model, where the arrows represent the \( R_{\text{define}} \) relations, and the dashed ovals represent the scope of the \( R_{\text{develop}} \) relations. First of all, this figure can promptly address the issue about “who is responsible for what”. For example, in the offering of their courses, instructors are also required to develop the relations of learning outcomes with grading components and graduate attributes.
or workshops for instructors and also standardization for some core courses.

Also, it is observed that the role of ADMIN is “separated” into defining the program courses (i.e., $R_{\text{define}}(\text{ADMIN}, PC)$) and developing the attribute indicators (i.e., $R_{\text{develop}}(\text{ADMIN}, AI)$) and their relations (i.e., $R_{\text{develop}}(\text{ADMIN}, R_{\text{refine}}(g_{a_i}, a_{i_j}))$). This may suggest some reorganizations of the contents and stakeholders’ responsibilities (e.g., associate attributes indicators with program courses). Without the graph-based model, administrators may experience difficulty in managing the information but do not know exactly why it is that (e.g., the possible disjoint between AI and PC on the graph).

Fig. 3. Stakeholders’ scopes with the content model.

3.2. Relations between Stakeholders and the Integration model

While the previous sub-section discusses how one type of stakeholders is related to some contents via $R_{\text{define}}$ and $R_{\text{develop}}$, this sub-section further specifies how stakeholders are related to one another. In this work, two types of relations between stakeholders are set: (1) regulate and (2) report. By the relation “regulate”, one stakeholder takes the managerial role to regulate another stakeholder, while the relation “report” indicates another direction, specifying one stakeholder is reporting to another stakeholder. Let $R_{\text{regulate}}(a, b)$ denote that stakeholder ‘$a$’ regulates stakeholder ‘$b$’, and let $R_{\text{report}}(b, a)$ denote that stakeholder ‘$b$’ reports to stakeholder ‘$a$’. In our context, accreditors regulate administrators on how the information of graduate attributes should be reported. Also, administrators regulate instructors on the collection and analysis of the graduate attribute data. These can be denoted as follows.

- $R_{\text{regulate}}(\text{ACC}, \text{ADMIN})$ and $R_{\text{report}}(\text{ADMIN}, \text{ACC})$
- $R_{\text{regulate}}(\text{ADMIN}, \text{INST})$ and $R_{\text{report}}(\text{INST}, \text{ADMIN})$

Notably, $R_{\text{regulate}}(a, b)$ implies $R_{\text{report}}(b, a)$ in our context. As specified earlier (e.g., Fig. 3), each stakeholder has their own scope of contents via the relations $R_{\text{define}}$ and $R_{\text{develop}}$. By taking $R_{\text{regulate}}$ and $R_{\text{report}}$, we can derive two general statements below related to our context.

- Statement 1: $R_{\text{regulate}}(a, b)$ and $R_{\text{define}}(a, c)$. This means that stakeholder ‘$a$’ defines content ‘$c$’ to regulate stakeholder ‘$b$’.

- Statement 2: $R_{\text{report}}(b, a)$ and $R_{\text{develop}}(b, c)$. This means that stakeholder ‘$b$’ develops content ‘$c$’ to report to stakeholder ‘$a$’.

Figure 4 illustrates the integration of the stakeholder and content models. This figure helps us to identify the relevant statements of the context. For example, the instances of Statement 1 are listed below.

- $R_{\text{regulate}}(\text{ACC}, \text{ADMIN})$ and $R_{\text{define}}(\text{ACC}, \text{GA})$: Accreditors define graduate attributes to regulate administrators.
- $R_{\text{regulate}}(\text{ADMIN}, \text{INST})$ and $R_{\text{define}}(\text{ADMIN}, PC)$: Administrators define program courses to regulate instructors.

Further, the instances of Statement 2 are listed below.

- $R_{\text{report}}(\text{ADMIN}, \text{ACC})$, $R_{\text{develop}}(\text{ADMIN}, \text{AI})$ and $R_{\text{develop}}(\text{ADMIN}, R_{\text{refine}}(g_{a_i}, a_{i_j}))$: Administrators report to accreditors with the information of attribute indicators and the relevant relations.
- $R_{\text{report}}(\text{INST}, \text{ADMIN})$, $R_{\text{develop}}(\text{INST}, \text{LO})$, $R_{\text{develop}}(\text{INST}, \text{GC})$, $R_{\text{develop}}(\text{INST}, R_{\text{measure}}(l_{\text{ok}, p}), g_{\text{ck}, a_i})$ and $R_{\text{develop}}(\text{INST}, R_{\text{associate}}(g_{a_i}, l_{\text{ok}}))$: Instructors report to administrators with the information of learning outcomes and gradating components and the relevant relations.

In sum, the integration model in Fig. 4 consists of five types of contents (i.e., $\text{GA}, \text{AI}, \text{PC}, \text{LO}, \text{GC}$), three types of content relations (i.e., $R_{\text{define}}, R_{\text{associate}}, R_{\text{measure}}$), three types of stakeholders (i.e., $\text{ACC}, \text{ADMIN}, \text{INST}$), two types of stakeholder relations (i.e., $R_{\text{regulate}}, R_{\text{report}}$) and two types of stakeholder-content relations (i.e., $R_{\text{define}}, R_{\text{develop}}$). While the information of individual portions (e.g., how GC measures LO) can be relatively simple, this model can become complex in pursuit of the consistency among different information pieces. In this view, this model can be used as the common ground for different stakeholders to clarify their intents and interpretations of the information. Besides, this model can support the development of the tools to support the accreditation process, and it will be discussed in the next section.
4. APPLICATIONS

4.1. Analysis of the COT and the ICDT

To better bridge the abstract graph-based model to the practicing COT and ICDT, we first provide an example to illustrate the actual use of the COT and the ICDT in a course. Then, we compare the information conveyed in the COT and the ICDT and the graph-based model and provide suggestions for further improvements.

Figure 5 shows an abbreviated sample of the COT for the course discussed in Section 2.3. This sample consists of three main parts. While the first part is the list of learning outcomes, the second and third parts show their mappings with graduate attributes and grading components, respectively. Notably, the traditional design of a course outline covers the first and third parts. With the additional second part, the quantitative data of grading components can be linked to the graduate attributes.

The ICDT user can generate the required mapping with the COT and the ICDT in view of the educational intents and collected grades in view of the educational intents and are not based on actual class performance in order to preserve confidentiality. In practice, an instructor submits the grading sheet that contains the grades of students in view of each grading component (i.e., the first “Assessment” column in Fig. 6). To support the accreditation reporting, the information of learning outcomes and graduate attributes (i.e., the “LO” and “GA” columns in Fig. 6, respectively) is mapped to grading components. As a result, the ICDT user can generate the five-bin distribution for graduate attribute(s). In this example, Figure 6 shows the five-bin distribution on the graduate attribute #8 “professionalism”, where the course assesses this attribute based on a question from Quiz #2 and selected questions from Final Exam.

With reference to the graph-based model in Fig. 4, both the COT and the ICDT are used to regulate instructors (i.e., \( R_{\text{regular}}(\text{ADMIN}, \text{INST}) \)), who are responsible for the following information of the \( k \)th program course (i.e., \( p_c_k \)).

- \( LO_k \) and \( R_{\text{refine}}(p_c_k, \text{lo}_k) \): Define a set of learning outcomes to refine the \( k \)th program course. In addition to the traditional course description, learning outcomes are intended to further specify what skills students need to learn in a course. This supports the educational intent from mainly knowledge understanding to skill development.

- \( R_{\text{associate}}(\text{ga}_i, \text{lo}_k) \): Associate each learning outcome with some graduate attributes. This mapping further clarifies how well a course is training our students in view of graduate attributes. If a learning outcome cannot be easily mapped to any one of graduate attributes, it implies that this learning outcome may not be directly contributive to the achievement of graduate attributes (but it can still be valuable to the training of engineering students). Alternately, if a learning outcome is mapped to many graduate attributes, it may not be effective to specifically train students in some particular aspects, not a good sign in education.

- \( G_{c_k} \) and \( R_{\text{measure}}(\text{lo}_k, \text{gc}_{c_k}) \): Define a set of grading components to measure the students’ performance in view the learning outcomes of the \( k \)th program course. While this grading aspect is common in a course outline, the required mapping needs to make the grading activities explicit to the context of learning outcomes. Similar to the above mapping concept, if a grading component cannot be easily mapped to any one of learning outcomes, it may not be easy to interpret the collected grades in view of the educational intents and
even graduate attributes. Also, if a grading component is mapped to many learning outcomes, it may be required to decompose this grading component into several parts to make the dimensions of grading clearer. To illustrate, we reuse the graph-based content model from Fig. 3 to highlight the domain of the COT and the ICDT in Fig. 7. As observed, both the COT and the ICDT do not particularly cover the aspect of attribute indicators (i.e., $AI$). Notably, attribute indicators are intended to refine graduate attributes. Once $LO$ is associated with $GA$ (i.e., $R_{associate}$) in the COT and the ICDT, it may cause the consistency issue between $AI$ and $LO$. Thus, one suggested revision is to remove the link between $LO$ and $GA$ (i.e., remove $R_{associate}(ga_i, lo_{pk})$) and add the link between $LO$ and $AI$. The new link is denoted as $R_{associate}(ai_{ij}, lo_{pk})$, which means the $j$th learning outcome of the $k$th program course is associated with the $j$th attribute indicator of the $i$th graduate attribute.

![Fig. 7. Domain of COT and ICDT with the graph-based content model.](image)

Arguably, $R_{associate}(ai_{ij}, lo_{pk})$ should be the most complex and detailed relation that needs to be set in the information system. Once it is set, the connections between grading components and graduate attributes can be established via learning outcomes (mostly understood by instructors) and attribute indicators (mainly interpreted by accreditors). In practice, we consider that the setup of $R_{associate}(ai_{ij}, lo_{pk})$ will be challenging. To support the operations, when defining the program courses (i.e., $R_{define}(ADMIN, PC)$), administrators may define a minimum set of learning outcomes, attribute indicators, and their associations per each program course. In turn, administrators do not need to change such information annually, but any changes can be treated as a formal process for continual improvements.

### 4.2. Adoption of Quality Function Deployment

Quality function deployment (QFD) is a well-developed method to support companies in quality management [8, 13]. Within QFD, one recognized tool is the house of quality, and one of its main functions is to build the connections between customers (expressed as customer requirements) and engineers (expressed as engineering characteristics). The essence of such connections is to ensure that the engineering efforts are geared toward the actual needs of customers. To make QFD practical, it is often adopted to a specific format so that practitioners can use it readily in their applications, and it is the intent of this section.

Program accreditation can be viewed as a quality assurance process. While accreditors express the quality requirements of engineering graduates, administrators and instructors are in the position to show how the engineering programs can train students to the prescribed quality. Based on the graph-based model of this paper, Figure 8 shows a sample QFD form for the $i$th graduate attribute (i.e., $ga_i$), which has four attribute indicators (shown the left column). By analogy, this represents the “customer requirements” in the context of the accreditation. Then, three program courses (i.e., $pc_1, pc_2, pc_3$) are listed on the right-side columns with their learning outcomes (e.g., $lo_{1,1}$). These program courses and their learning outcomes indicate how the engineering program provides training for the $i$th graduate attribute. In particular, each cross ‘X’ in the QFD form indicates a learning outcome is responsible for an attribute indicator (e.g., $lo_{1,2}$ is responsible for $ai_{1,2}$ in Fig. 8). Then, this QFD matrix form allows stakeholders to indicate and comment on the mapping between attribute indicators and learning outcomes.

While Fig. 8 shows only one graduate attribute, this matrix form can be expanded for twelve graduate attributes with all program courses. This expanded matrix can serve as a giant map that supports accreditors and administrators to navigate both the high-level and detailed information. At the high level, we can check how each graduate attribute is covered by a subset of program courses. At the detailed level, we can check how an attribute indicator is addressed by a program course in view of learning outcomes. In practice, administrators can use this giant map as the control document to examine the following issues.

- Is there any graduate attribute or attribute indicator that is weakly presented in program courses?
- Is there any program course or learning outcome that is not contributive to graduate attributes?

![Fig. 8. QFD form on the $i$th graduate attribute.](image)
grading components indicate how well these learning outcomes are addressed.

<table>
<thead>
<tr>
<th>pc1</th>
<th>lo1,1</th>
<th>gc1,1</th>
<th>gc1,1</th>
<th>gc1,1</th>
<th>gc1,1</th>
</tr>
</thead>
<tbody>
<tr>
<td>lo1,2</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>lo1,3</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>lo1,4</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Fig. 9. QFD form at the instructor level.

Strictly speaking, the QFD matrices discussed here are replicating the same information conveyed in the graph-based model in a different format. Notably, QFD has been applied in quality management to support benchmarking and continual improvements [13]. Thus, this development can potentially support the relevant accreditation activities in a similar systematic manner.

4.3. Software Proposal for Content Management

As discussed in this paper, the information involved in the accreditation can be intensive. The graph-based model can be used to support the development of a software tool to manage and analyze the information contents related to the accreditation. In this context, the software tool can have two main functions. The first one is to integrate the incoming information from the COT and the ICDT to form a consistent inventory of the information contents based on the graph-based model. Secondly, it can analyze the information stored in the system and generate different types of reports to serve various needs of stakeholders. Figure 10 overviews the basic functions to this software development.

Once the structure of contents with quantitative measures is built, the software can generate some useful information automatically in principle. Firstly, the curriculum map (required by CEAB) can be generated by getting a high-level mapping between GA and PC. Secondly, the QFD matrices discussed in Section 4.2 can be generated at different resolution levels. Thirdly, the software can formulate a statement on the measurements of graduate attributes based on what being provided by the ICDT (e.g., Fig. 6). The formulated statement can be expressed as follows, where the underlined terms are variables.

- Twenty percent of students achieve the excellent level of GA #7: Communication skills in view of some attribute indicators. This measurement is supported by the course with relevant learning outcomes and grading components.

The proposal of the software tool development is still at the early stage. What seems to be promising to us is that the graph-based model can provide a stable structure of the information contents. It can help to standardize the information flows in a consistent and regular manner. At the same time, the relevant analysis and reporting on the performance of graduate attributes can be discussed in a standard context. In such a way, while the collected information on graduate attributes can be more transparent and reliable, administrators and instructors can devote more efforts to the continual improvements.

5. SUMMARY AND FUTURE WORK

The outcome-based accreditation criterion has introduced challenges associated with collecting and analyzing the graduate attributes information reliably and meaningfully. In this regard, this paper proposes a graph-based model that specifies the information contents and their relational dependencies with stakeholders in the accreditation process. The graph-based model has classified five types of information contents, three types of stakeholders, as well as three types of content-content relations, two types of stakeholder-stakeholder relations and two types of stakeholder-content relations. To illustrate the applications, the graph-based model has been used to review the course outline template (COT) and the Integration Course Design Tool (ICDT) and propose the implementations of quality function deployment (QFD) and the software tool. In the future work, we will investigate the potential to define the relations between attribute indicators and learning outcomes for individual courses as the foundational structure of an engineering program. Then, continual improvements can be interpreted as a systematic procedure to improve such a structure. At the same time, management and software tools will be further refined and implemented to support this development.
Acknowledgements

We acknowledge our supporting staff, especially Adrianne Hrynyszyn and Robyn Mae Paul, for their effort and feedback so that we can get more practical views in the data collection process associated with graduate attributes. Also, the discussions in the Program Quality Assurance Committee, led by Jeff Pieper, have helped nurture the ideas of this paper.

References