

A Meta-Model for Strategic Educational Goals

Mohammad Alhaj¹, Ashraf Sharah²

Computer Engineering Dept¹

Electronics and Communications Engineering Dept²

Al-Ahliyya Amman University

Amman, Jordan

Abstract—Metamodeling technique is adopted widely in different fields related to software and system engineering. A meta-model represents the abstraction of a detailed design at multiple level. It is used in any structured environment ruled by a certain constraints and obligations and instantiate different platform specific domains from a single platform independent domain. This paper proposes a new model-driven approach for generating and analyzing automatically the outcome measures of strategic educational goals model. A new meta-model augmented with arithmetic semantics is created for Strategic Educational Goals where a set of outlines defines the enhancement framework of an academic organization. The vision, mission, program educational objectives and student outcomes are the four common strategic educational goals. These Goals support the performance roadmap to measures the institution situation and progress. The proposed meta-model is used to evaluate the strategic educational goals in a formal way, improve the continuous improvement process in academic organizations and allows the assessment at different level of management.

Keywords—Model-driven engineering; meta-model; goal model; core modeling framework; object constraint language; strategic educational goals

I. INTRODUCTION

In recent software development technologies, model-driven engineering (MDE) is used to focus on exploiting and handling conceptual models of all aspects related to a specific domain and raise the level of abstraction. The purpose is to provide a systematic and formal approach that would make the software development process more adaptable, portable and reusable. It also would improve the interaction between the stakeholders and reduce the effort and complexity of the software development [1]. Metamodeling is one of common techniques of MDE that allows the modeller to create multiple concepts of a model and instantiate different platform specific domains from a single platform independent domain. It is adapted in various areas in software engineering and hardware industries. It can be used in any structured environment ruled by a certain constraints and obligations. A meta-model is a framework where a set of entities, relationships, rules and constraints are used to describe a modeling domain. In other word, it is a model of a model or an abstract model of a concrete model. We used meta-model framework in the academic field to provide a systematic method in modeling and analyzing the organization status and evolution.

Academic institutions are promoting for strategic educational goals (SEG) to frame their current situation and

future progress and enhancement. Fig. 1 presents the SEG as a set of outlines and statements that define the roadmap of an institution and shapes its character. At the institution level, the vision statement describes the organization ultimate achievements and gives purpose of its continuation. The mission statements are then defined at multiple levels of management: institution, faculty and department and they outline the organization overall objectives. From the mission statements, the program educational objectives (PEOs) are created to describe what graduates are expected to achieve within next years of graduation. The student learning outcomes (SOs) are stemmed from the PEOs to specify the knowledge and skills the student will demonstrate after completing a certain course.

The performance outputs in SEG environment are measured using two assessment tools: direct assessment where the tool measures the knowledge and skills of students through assigned task and exams; and indirect assessment where the tool measures the implicit qualities with respect to a group of people using surveys and discussion groups. Stakeholders and constituencies are involved in outlining, evaluating and improving SEG at multiple-level of management. Two kind of constituencies are recognized: internal constituencies such as academic instructors, registered students, focus groups, academic councils, officers, and administrators; and external constituencies, such as industrial advisory board (IAB), alumni and students' parents.

In this paper, we propose a new meta-model for generating and analyzing automatically the outcome measures of SEG model. The research uses a common modeling framework and code generation from eclipse plugins, called EMF, to build the SEG meta-model and generate the code of meta-classes, meta-types and arithmetic semantics that would allow analyzing the performance measures of the SEG elements. We validate our proposed research using an ongoing project developed in a local university.

The paper is organized as follows: Section II presents the background and related work; Section III presents the SEG architecture, meta-data and method used for evaluating the output measures; Section IV introduces the SEG meta-model Development; Section V presents a case study of generating and evaluating SEG model; Section VI presents the conclusion and future work.

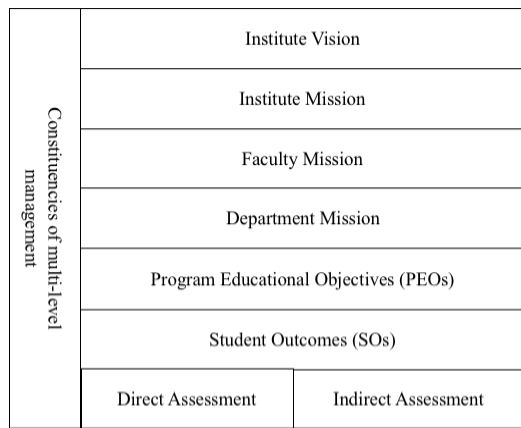


Fig. 1. SEG Block Diagram.

II. BACKGROUND AND RELATED WORK

Metamodeling is model-driven engineering method that has been adopted widely in different fields related to software and system engineering. A meta-model represents the abstraction of a detailed design at multiple level. It is used to instantiate multiple models that always adapt its rules and constraints. A meta-model has been adapted at different recent literature researches such as mathematical relation [2], algorithm characterizing input and output relations [3], business processes [4] and neural networks [5]. There is also Meta-Object Facility (MOF) [6] the common meta-model architecture in software engineering. MOF is a metamodeling architecture, owned by object management group OMG [7], that supports a type system for objects in the Common Object Request Broker (CORBA) architecture [8]. It consists of four layers: the data layer (M0), the model (M1), the meta-model (M2) and the meta-meta-model (M3). Also, the Object Constraint Language (OCL) is used to declare the constraints that specify the invariant conditions of a meta-model [9]. Several tools support a modeling framework and code generation to build meta-models. One of the common tools available in the market is a plugin called eclipse modeling framework (EMF) [10]. EMF is graphical editor that allows the modellers to build a meta-model using the UML class diagram. The produced meta-model is used to generate a source code that comes as a set of java classes and interfaces. The generated source code can be used to instantiate the meta-model to produce a concrete model.

Researchers have created various meta-models at different aspects of research projects. Roy et al. in [11] propose an eclipse plugin graphical editor for goal and scenario modeling called jUCMNav. The tool is created based on a metamodel for a User Requirement Notations (URN) language where two modeling languages are integrated: the Goal-Requirement Language (GRL) and Usecase Map (UCM). Schieferdecker in [12] proposes a metamodel of Testing and Test Control Notation (TTCN-3); and reviews the realization of the TTCN-3 metamodel on modeling tools. While in Frank [13] presents a multi perspective enterprise modelling (MEMO) which describes the semi-formal concepts to specify various graphical modelling languages within the MEMO framework. Djuric et al. in [14] uses the four layers of model driven architecture (MDA) standards to present the ontology definition metamodel

(ODM). While Richters and Gogolla in [15] propose a metamodel for Object Constraint Language (OCL), an extension for UML constraints.

Also, several researches have followed the SEG outlines and statements to predict for the future improvements of an organization. Alhaj in [16] proposes a model-based technique to generate goal models for the learning outcomes, augmented with quantitative indicators. The models will improve the assessment process and evaluate the learning components in a formal way. Prentice and Robinson in [17] uses the service learning a that combines the community service with the academic instruction to enhance the student learning outcomes. While Maher in [18] studies the effect of learning outcomes in higher education and their implications on curriculum design. Also, Duque and Weeks in [19] propose a conceptual model and supporting tool to assess the learning outcomes of undergraduate students and satisfaction with their program.

In summary, it is obvious that none of the works above have used a generated SEG metamodel to perform assessment for the learning outcomes and objectives in academic institutions. The proposed approach is used to evaluate the strategic educational goals in a formal way, improve the continuous improvement process in academic organizations and allows the assessment at different level of management.

III. THE ARCHITECTURE AND METADATA OF SEG

In this section, we introduce the metadata of the SEG described as in Fig. 2. The metadata provides information and the interrelationship between the elements in SEG data domain. It also defines the global view of the modeling domain which helps to build the SEG metamodel.

The Organization meta-element defines the entity that embraces all the meta-elements within the meta-data. It comes at the top of the meta-data and represents different types of academic organization structures, such as institution, faculty, department and program. The Vision meta-element of an institution organization structure represents a statement that outlines the organization fundamental successes and used to define an institution Mission meta-element where statement outlines the organization overall objectives. The Vision meta-element are only defined for an institution structure type and it does not exist with other structure types. Below the institution Mission meta-element, there are sub Missions for the faculty, department and program structures. The sub Missions are used to stem the generation of multiple of program educational objects (PEO) meta-elements. The PEO meta-elements are assessed by multiple of Indirect Assessment meta-elements. The course outcomes for each Course is defined by the student outcomes (SOs); and it can be assessed by Direct Assessment and Indirect Assessments meta-elements. The Organization also contains several kinds of Constituencies meta-element to manage and monitor its internal policies and bylaws.

A. Methods of Evaluating and Assessing the Output Measures

The analysis aspect of the metadata is performed by evaluating the performance measures produced by the direct/indirect assessment tools. These measures are then accumulated and propagated in the metamodel to reflect compliance of SEG meta-elements. The data-model provides

five-level of assessment where at each level some of the SEG meta-elements are involved as in Table I. In this section, we describe the arithmetic semantics of evaluating the output measures of the Assessment Tools, SOs, PEOs, Missions and Vision.

B. Evaluating the Performance Measures using Assessment Tools

The evaluation value of the SOs are calculated using the direct/indirect assessment tools. Fig. 3 illustrates the approach of calculating evaluation values of SO. For a number of students= J, the average grade of students who performed a specific assessment tool is calculated as:

$$Average(Grade) = \frac{\sum_{j=1}^J Student_Grade_j}{J}$$

The evaluation value of SO that are influenced by number (N) of assessment tools:

$$Evaluation(SO_k) = \sum_{n=1}^N \frac{Average(Grade_n) * Weight(SO_k)}{Max(Assessment_Grade_n)} * 100 \quad (1)$$

Where, SO_k defines a set of student outcomes, such that $k = \{1...K\}$, and K is the number of influenced SOs.

$Max(Assessment_{Grade})$ is the highest grade of an assessment tool.

Weight(SO) is the relative contribution weight of an assessment tool (AT) to the upper SO, that ranges from 0 to 100.

C. The Accumulated Output Measures of Direct and Indirect Assessment Tools

The SEG meta-elements of the metamodel are correlated in a way that their performance measures produced by the direct/indirect assessment tools are propagated to the upper performance measures of meta-elements. Fig. 1 describes the SEG block diagram, where the evaluation values of the SOs calculated as in equation (1) using the direct/indirect assessment tools. The evaluation values of SOs are then propagated to calculate the evaluation values of the PEOs, Mission and Vision using the relative Weight(PEO) and relative Weight(Mission) respectively.

Fig. 4 illustrates the general approach of calculating evaluation values of the upper meta-elements, i.e. PEOs, Mission and Vision.

$Evaluation(Upper_k) =$

$$\frac{Evaluation(Lower_1) * Weight_{k1} + \dots + Evaluation(Lower_N) * Weight_{kN}}{Weight_{k1} + \dots + Weight_{kN}}$$

$$Evaluation(Upper_k)$$

$$= \sum_{n=1}^N \frac{Evaluation(Lower_n) * Weight_{kn}}{Weight_{kn}} \quad (2)$$

Where, k is the number of influenced meta-elements.

Weight is the relative contribution weight of lower to the upper meta-element that ranges from 0 to 100.

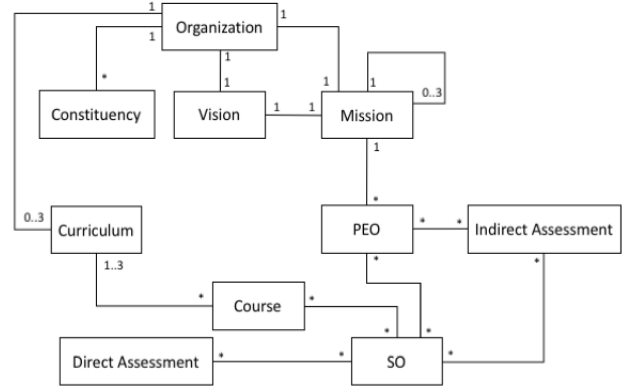


Fig. 2. The Metadata of Strategic Educational Goals (SEG).

TABLE I. TOP-DOWN VIEW OF SEG METADATA

Assessment Level	Performance Measures	Assessment Tools
Institution	Mission, Vision	Indirect
Faculty	Mission	Indirect
Department	Mission	Indirect
Program	PEOs, Mission	Indirect
Curriculum	SOs	Direct/Indirect
Course	SOs	Direct/Indirect

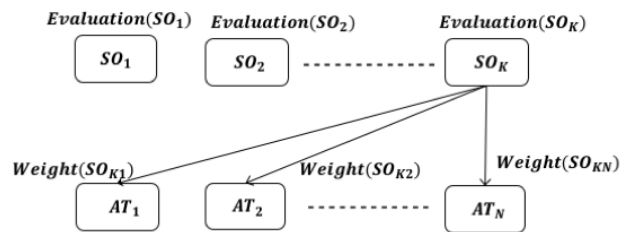


Fig. 3. Calculating Evaluation Values of SO using Direct/Indirect Assessment Tools.

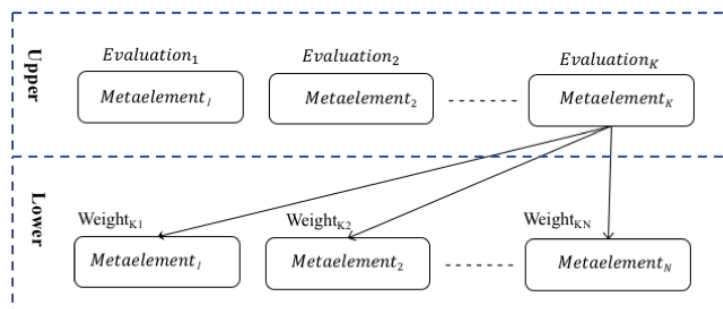


Fig. 4. Calculating the Evaluation Values of the upper Meta-Elements Propagated from Lower Meta-Elements.

IV. SEG METAMODEL DEVELOPMENT

We used an eclipse plugin called Eclipse Modeling Framework (EMF) [10] to build the SEG metamodel as in Fig. 5. EMF is a modeling framework and code generation plugin that supports building modeling tools through a set of java classes generated automatically from a specific metamodel. With three levels code generation, EMF provides all java interfaces and classes for modeling, adopting and editing models. The SEG metamodel in Fig. 5. is generated based on the meta-data described in the previous section. The majority of meta-data elements, meta-type and the relationship are mapped directly.

The Organization meta-class comes at the top of the metamodel and contains all the other meta-classes. A composition relationship \blacktriangleleft is used to indicate a restricted containment between the Organization meta-class and the two meta-classes: the Link_Element and Node_Element. The cardinality between the meta-classes varies from 0..1, 0..*, 1..1, and 1..* depends on the number of meta-instances involved. The Organization meta-class defines four attributes: id, name and description and orgType which represents a list of Organization_Enum types such as institution, faculty, department and program. The Link_Element meta-class is used to connect the node elements together. It has two attributes: the id and the contribution_weight that ranges from 0 to 100 and defines the relative contribution of a meta-class to the other meta-class at higher levels. The Node_Element meta-class is a generalized class for most of the meta-classes in SEG metamodel described as follows:

- The Vision meta-class with three attributes: id, statement and evaluation_value which defines the Vision assessment amount. The Vision meta-class has a 1..1 relationship with the Mission meta-class.

- The Mission meta-class with three attributes: id, statement and evaluation_value which defines the Mission assessment amount. The Mission meta-class has a 0..3 relationship with itself, and a 1..* relationship with the PEO meta-class.
- The PEO meta-class with three attributes: id and definition, and evaluation_value which defines the PEO assessment amount. The PEO meta-class has a 0..* relationships with SO meta-class and Indirect_Assessment meta-class.
- The SO metadata with three attributes: id, definition and evaluation_value which defines the SO assessment amount. The SO meta-class has a 0..* relationships with the Direct_Assessment and Indirect_Assessment meta-classes.
- The Assessment meta-class is a generalized abstract for two meta-classes: Direct_Assessment and Indirect_Assessment. Both meta-classes define six attributes: id and name, max_value, avg_value, min_value, that define the maximum, average and minimum grades of the assessment tools, and evaluation_value that defines the actual amount of the assessment tool (Average students' marks).
- The Constituency meta-class with three attributes: id, name and description.

In additional of generating SEG metamodel, we used object constraint language (OCL) to declare the rules and expressions that specify the invariant conditions of the SEG metamodel. A sample of the conditions, described in Table II, are extracted from the SEG metadata and arithmetic semantics described in the previous section. Eclipse Xtext editor allows the modelers to test and validate the OCL constraints; then these constraints can be embedded into the SEG metamodel.

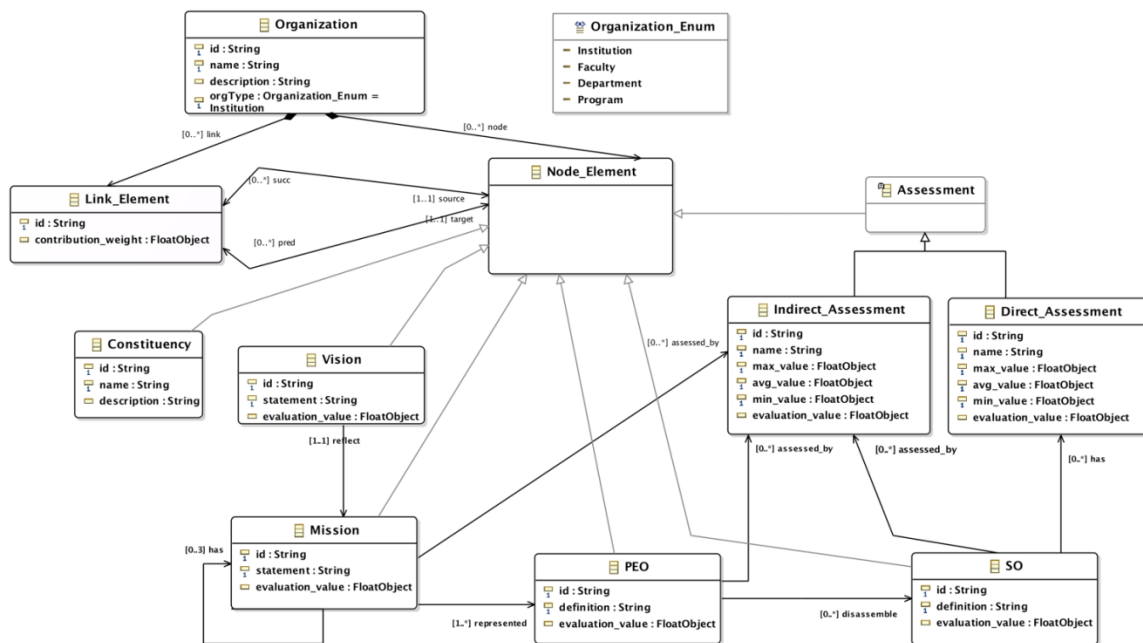


Fig. 5. SEG Metamodel using Eclipse EMF Plugin.

TABLE II. A SAMPLE OF OCL CONSTRAINTS OF SEG METAMODEL

Instance	Condition	OCL constraint
Organization	Only the Organization of type "Institution" has a Vision instance	context Organization inv: if self.orgType<>'institution' then context::instanceClassName("Vision") -> Set(null) endif
Mission	Attribute: $0 \leq \text{evaluation_value} \leq 100$	context Mission inv: self.evaluation_value => 0.0 and self.evaluation_value <= 100.0
PEO	Attribute: $0 \leq \text{evaluation_value} \leq 100$	context PEO inv: self.evaluation_value => 0.0 and self.evaluation_value <= 100.0
SO	Attribute: $0 \leq \text{evaluation_value} \leq 100$	context SO inv: self.evaluation_value => 0.0 and self.evaluation_value <= 100.0
Direct_Assessment	Attribute: $1 \leq \text{max_value} \leq 100$	context Direct_Assessment inv: self.max_value >= 1 and self.max_value <= 100 self.avg_value <= max_value self.avg_value >= 0 and self.avg_value <= 100 self.min_value <= avg_value self.min_value >= 0 and self.min_value <= 100 self.evaluation_value >= self.min_value and evaluation_value <= max_value
	Attribute: avg_value <= max_value and $0 \leq \text{avg_value} \leq 100$	
	Attribute: min_value <= avg_value and $0 \leq \text{min_value} \leq 100$	
	Attribute: min_value <= evaluation_value <= max_value	
	Attribute: average_value <= maximum_value and $0 \leq \text{average_value} \leq 100$	
	Attribute: minimum_value <= average_value and $0 \leq \text{minimum_value} \leq 100$	
Indirect_Assessment	Attribute: $1 \leq \text{max_value} \leq 100$	context Direct_Assessment inv: self.max_value >= 1 and self.max_value <= 100 self.avg_value <= max_value self.avg_value >= 0 and self.avg_value <= 100 self.min_value <= avg_value self.min_value >= 0 and self.min_value <= 100 self.evaluation_value >= self.min_value and evaluation_value <= max_value
	Attribute: avg_value <= max_value and $0 \leq \text{avg_value} \leq 100$	
	Attribute: min_value <= avg_value and $0 \leq \text{min_value} \leq 100$	
	Attribute: min_value <= evaluation_value <= max_value	
	Attribute: average_value <= maximum_value and $0 \leq \text{average_value} \leq 100$	
	Attribute: minimum_value <= average_value and $0 \leq \text{minimum_value} \leq 100$	
Link_Element	Attribute: $0 < \text{contribution_weight} \leq 100$	context Link_Element inv: self.contribution_weight > 0 and self.contribution_weight <= 100

The SEG metal model and its related OCL constraints are used to generate SEG model code. Eclipse EMF plugin supports generating automatically java classes and interfaces of the meta-classes and their relationships, as in Fig. 6. The classes and interfaces have the attributes and functions needed to create and edit the SEG model and present them in XML

format. We also need to create manually the code of the arithmetic semantics used for analyzing the performance measures of the SEG elements. A sample of the algorithm used for implementing equation 1 in previous section is described below.

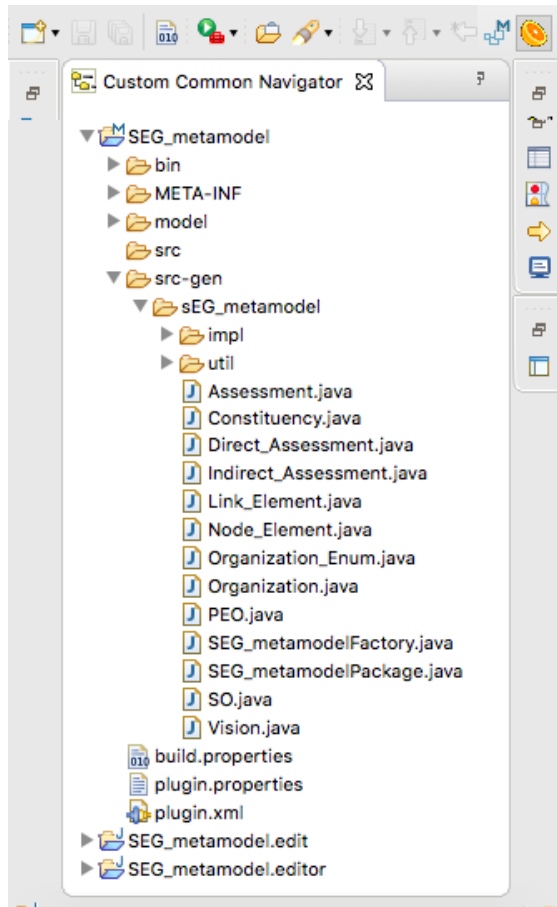


Fig. 6. EMF Java Model Classes of SEG Metamodel.

// Algorithm of equation (1): Calculating evaluation values of SO using direct/indirect assessment tools

```

start
for each SO in the SEG model
for each Assessment in the SEG model
Total_Grade= 0, Total_Weight= 0
if instances of Link_Element = true
Total_Grade= Total_Grade + Assessment.evaluation_value
* Link_Element.Weight
Total_Weight= Total_Weight + Link_Element.Weight
end if
end
Set Evaluation_value of SO (Total_Grade/Total_Weight)
End
    
```

V. A CASE STUDY OF GENERATING SEG MODEL

In this section, we introduce a sample of our proposed SEG metamodel. The sample is a SEG model developed at Faculty of Engineering in [20] as a part of an ongoing project. The project aims to maintain continuous improvement of academic programs at the Faculty of Engineering and to qualify the

programs to gain ABET accreditation [21]. Five programs are part of the project: Computer Engineering, Civil Engineering, Communications and Electronics Engineering, Electrical Engineering and Medical Engineering. During the continuous improvement process, the SEG elements are assessed using several direct and indirect assessment tools. We used our generated java code of SEG metamodel to create a sample of SEG model.

In this project, we used a 1-5 scale for contribution_weight attribute of the Link_Element instance. The weight reflects the relative contribution of a model element with respect the upper node elements. Several group decision approaches and techniques found in [22] and can be used for assigning the relative contribution_weight to each Link_Element. We used a Round-Table Discussion and Consensus (RTD&C) approach, where focus groups are gathered in a discussion form. Groupings of related elements contained in models are put up on a screen, and the focus group members are asked to discuss and assign relative contribution_weight to each Link_Element in each grouping. Fig. 7 summarizes the values of the contribution_weight of our SEG model.

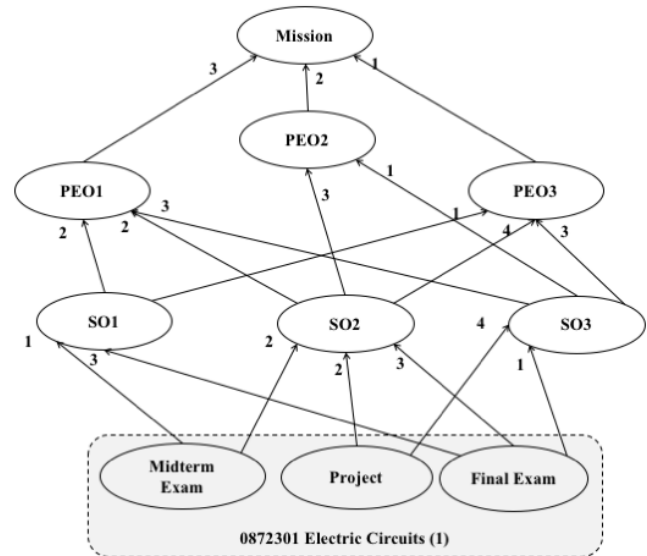


Fig. 7. The Relationship and Contribution_Values of the Case Study.

At the top level, the Organization instance with a name=“Electrical Engineering”, Description=“The electrical engineering program at the faculty of engineering” and orgType= program.

The Mission statement declares that “Excellence in the quality of the graduates academically and professionally to meet the requirements of the local and regional labor market and to keep up with the technological advancements in the field. Stimulate and strengthen the scientific research in Electrical Engineering” [20].

A sample of three PEOs contributes to the Mission by weights 3, 2 and 1, respectively. The definition of the PEOs are:

- “PEO1: Identify, analyze, formulate, and solve electrical engineering problems associated with the

workplace, both independently and in a multidisciplinary team environment”.

- “PEO2: Demonstrate commitment and progress in a continuous learning, professional development, and leadership”.
- “PEO3: Design electrical systems”.

There are three SOs contribute to the PEOs by weights that range from 1 to 4. The definition of the SOs are:

- “SO1: An ability to apply knowledge of mathematics, science, and engineering”.
- “SO2: An ability to identify, formulate, and solve engineering problems”.
- “SO3: An ability to use the techniques, skills, and modern engineering tools necessary for engineering practice”.

Based on a Course at electrical engineering program: “0872301 Electric Circuits (1)” taught by one internal Constituency: “Instructor 1”. The course is assessed by three Direct_Assessment tools which contribute by weights that range from 1 to 4. The Direct_Assessment tools are defined as follows:

- For 0872301 Electric Circuits (1):

a) name= ”Midterm Exam”, max_value= 30, avg_value= 18, min_value= 10, evaluation_value= 12.

b) name= ”Project”, max_value= 15, avg_value= 10, min_value= 5, evaluation_value= 7.

c) name= ”Final Exam”, max_value= 40, avg_value= 25, min_value= 18, evaluation_value = 22.

The SEG model of the case study is described in emf format (Fig. 8). Each element in emf format is represented in a single line. The parent node element is the Organization Electrical Engineering which embraces all other elements. The children *Node_Elements* are modeled based on details of the case study described before. The *Linked_Elements* are described at the end of the model with the *contribution_value* of each one.

A. Evaluation of Generated SEG Model

In this paper, we propose a SEG metamodel to allow the researchers at [20] to model the strategic educational goals at different levels of an academic organization and evaluate the performance measures of the SEG elements produced by the direct/indirect assessment tools. The metamodel is developed using EMF eclipse plugin that provide a modeling framework and code generation features.

Table III presents the analysis of the generated SEG model based on the evaluation_value attribute of elements. The evaluation values of the Program Mission, PEOs and SOs seems to be below the expectation. However, these evaluation values the SEG model is only the contribution of one course. The case study, in the previous section, is part of a bigger SEG model that contains more courses of the curriculum and all of them contribute to the SOs, PEOs and the Mission. So, the actual evaluation_values are within the satisfying range.

In the case study, we present a single iteration of obtaining the performance measures of SEG elements. This is definitely inconvenient in obtaining a meaningful results from performance measures of the SEG elements. In continuous improvement process, it is more reasonable to evaluate the performance of SEG elements and perform model analysis multiple times, in a kind of cycles where a single cycle represents one academic semester.

Also, several challenges were addressed during the case study practice due to large number of participated constituencies and lack of quality former performance measures. In such modeling approach, teams from different disciplines are required to meet periodically to discuss the modeling structure, define the modeling elements and relationship between them and assign the contribution_weight between the model elements. This might increase the chance of human error and increase the duration of becoming familiar with modeling approach. Another challenge caused by the textual format of the generated SEG model, which makes it difficult for the modelers to manage and trace it. This challenge will be addressed in the future work where a graphical editing feature will be added to the proposed metamodel using eclipse Graphical Editing Framework (GEF) plugin [23].

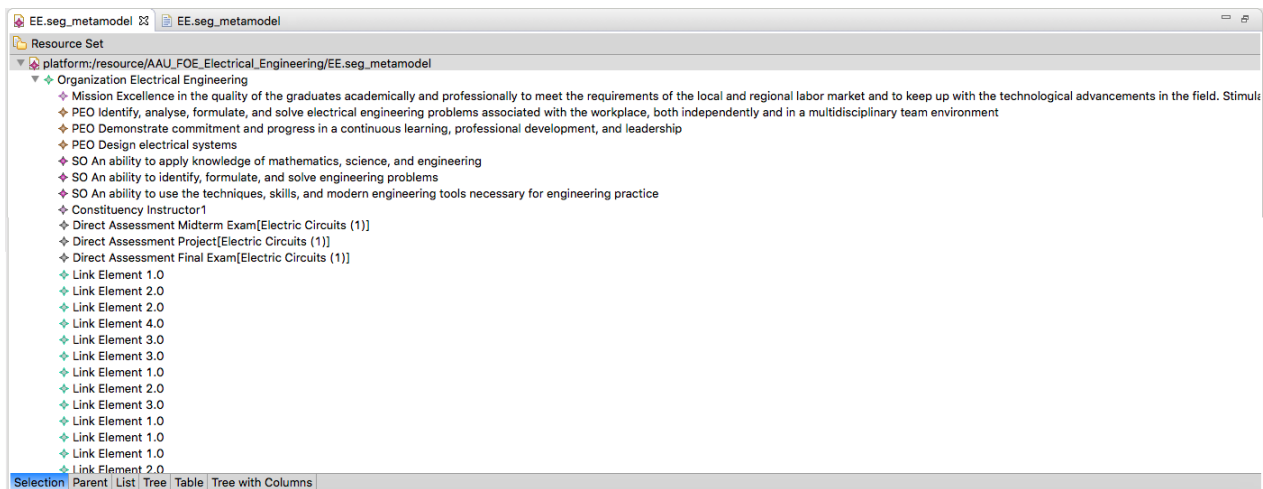


Fig. 8. SEG Model of the Case Study in EMF Format.

TABLE III. THE EVALUATION VALUES OF THE SEG ELEMENTS IN THE CASE STUDY

SEG element	Evaluation value (%)
Program Mission	11
PEO1	21
PEO2	26
PEO3	15
SO1	55
SO2	41
SO3	65

VI. CONCLUSION AND FUTURE WORK

The traditional paper-based approaches are using documents and spread sheets for evaluating the strategic educational goals. This might lead to inconvenient in analyzing the objectives and goals, lack of clarity, and subject to different interpretations by researchers. The paper proposes a novel model-driven approach for generating and analyzing automatically the outcome measures of SEG model. The research uses a common modeling framework and code generation from eclipse plugins called EMF to build the SEG metamodel and generate the code of meta-classes, meta-types and arithmetic semantics used for analyzing the performance measures of the SEG elements. We also validated our proposed research using an ongoing project developed in a local university.

As future work, we are going to extend our work by using eclipse GEF plugin. GEF is a graphical framework used to create the graphical view of the metamodel. We also will use the proposed metamodel on different case studies.

ACKNOWLEDGMENT

Our thanks to Faculty of Engineering ABET Committee at AAU for the help and support they provide during the accreditation process.

REFERENCES

[1] M. Werfs, "Model-driven Engineering – a promising approach for developing critical software applications", Republication of think creative 3.0, thinkcreative30.wordpress.com.
[2] D. Akehurst, S. Kent, "A Relational Approach to Defining Transformations in a Metamodel", The Unified Modeling Language. UML 2002, Lecture Notes in Computer Science, vol 2460, 2002.

[3] M. Bonte, A. van den Boogaard, J. Huetink, "A Metamodel Based Optimisation Algorithm for Metal Forming Processes", In: Advanced Methods in Material Forming, 2007.
[4] B. List B., B. Korherr, "An evaluation of conceptual business process modelling languages", Proceedings of the 2006 ACM symposium on Applied computing, pp 1532-1539, France, 2006.
[5] D. Lechevalier, S. Hudak, R. Ak, Y. T. Lee, S. Foufou, "A neural network meta-model and its application for manufacturing", IEEE International Conference on Big Data (Big Data), IEEE, 2015.
[6] OMG, Meta Object Facility, <https://www.omg.org/mof/>.
[7] OMG, Object Management Group, <https://www.omg.org/index.htm>.
[8] OMG, Common Object Request Broker Architecture, <https://www.omg.org/spec/CORBA/About-CORBA/>
[9] OMG, Object Constraint Language, <https://www.omg.org/spec/OCL/2.0/About-OCL/>
[10] Eclipse Modeling Framework (EMF), <https://www.eclipse.org/modeling/emf/>.
[11] J. Roy, J. Kealey, D. Amyot, "Towards Integrated Tool Support for the User Requirements Notation", System Analysis and Modeling: Language Profiles, Lecture Notes in Computer Science, vol 4320, 2006.
[12] I. Schieferdecker, G. Din, "A Meta-model for TTCN-3", Applying Formal Methods: Testing, Performance, and M/E-Commerce, Lecture Notes in Computer Science, vol 3236, 2004.
[13] U. Frank, "The memo meta-metamodel", CiteSeerX, 1998.
[14] D. Djuric, D. Gasevic, V. Devedzic, "Ontology Modeling and MDA", Journal of Object Technology 4(1):109-128, 2005.
[15] M. Richters, M. Gogolla, "A Metamodel for OCL", International Conference on the Unified Modeling Language, Lecture Notes in Computer Science, vol 1723. Springer, Berlin, 1999.
[16] M. Alhaj, "Towards model-based evaluation process of learning outcomes in academic institutions", 7th International Conference on Information and Education Technology (ICIET 2019), Aizu-Wakamatsu, Japan, March 29-31, 2019.
[17] M. Prentice, G. Robinson, "Improving Student Learning Outcomes with Service Learning", Higher Education, <https://digitalcommons.unomaha.edu/slcehighered/148>
[18] A. Maher A., "Learning Outcomes in Higher Education: Implications for Curriculum Design and Student Learning", The Journal of Hospitality Leisure Sport and Tourism 3(2), 2004.
[19] L. C. Duque, J. R. Weeks, "Towards a model and methodology for assessing student learning outcomes and satisfaction", Quality Assurance in Education 18(2):84-105, 2010.
[20] AAU, Al-Ahliyya Amman University, <https://www.ammanu.edu.jo/English/HomeP/Home.aspx>
[21] ABET, <http://www.abet.org/accreditation/accreditation-criteria/>
[22] O. Akhigbe, M. Alhaj, D. Amyot, O. Badreddin, E. Braun, N. Cartwright, G. Richards, G. Mussbacher, "Creating Quantitative Goal Models: Governmental Experience", International Conference on Conceptual Modeling, pp 466-473, 2014.
[23] Eclipse Graphical Editing Framework, <https://www.eclipse.org/gef/>