Published work: DOI: 10.1109/TALE56641.2023.10398291

Using Learning Objective-based Course Modeling for Complete Exercise Generation: From Course Material to an Aggregated Knowledge Representation

Deniz Schmidt
Institute for Technical Logistics
Hamburg University of Technology
Email: deniz.schmidt@tuhh.de

Max Uzulis
Center for Teaching and Learning
Hamburg University of Technology
Email: max.uzulis@tuhh.de

***Abstract*—Learning objectives are an essential part of teaching, but are rarely accounted for in digitization processes of courses. Instead of naively translating existing material into a digital form, they offer a chance to examine a course's didactic soundness and to create an aggregated knowledge representation. In this paper we present an applied method to digitally model courses based on learning objectives, coupled with a special exercise generator. This approach includes the extraction and categorization of learning objectives from course contents, the formal definition of subject matter and a topical overview of the course. The resulting model allows for a critical view on the didactics of the course through constructive alignment, while also serving as the basis for the exercise generator. Conclusively the effects of the method on the teaching experience, as well as the usability are discussed.**

***Keywords*— Learning Objectives, Course Modeling, Exercise Generation, Assessment**

I. Introduction

Engineering sciences play an important role in a modern and sustainable society. With this comes an increasing demand for high quality engineering education. One aspect of this education is to find effective and inclusive digitized learning solutions [1], [2]. To match the individual learning needs, stemming from different academic backgrounds and skill levels of students, we are developing an AI-based, complete exercise generator that is capable of generating individualized exercises, fitted to each student at every point of their learning progress [3]. This helps students in the field of engineering to practice on their own and at their own pace.

For such an exercise generator the digital modeling of courses is required. Digitization of paper-printed course material (books, lecture scripts, etc.) is a minimal solution to supply students with a modern learning experience, but it insufficiently represents or even disregards the underlying didactic structures of the courses themselves and the possibility to enhance digital teaching practices further.

As a solution this paper presents a learning objective based approach for the modeling of courses and its implications for the generation of individualized exercises.

A. Goal and Approach

For the overall goal to generate individualized exercises a digital representation of the courses is needed. This final knowledge model implies some technical necessities, but also allows for the embedding of didactic methods. Thus it should:

- represent courses in a user(teacher)-navigable and machine-readable way
- support the automatic creation of individualized exercises
- help teachers reflect upon the course structure and enhance the teaching experience

To achieve this we chose to document all learning objectives of a given course and map them onto a network of machine-readable, topical structured, atomic knowledge units. We call this an aggregated knowledge model. In of itself this is not a complete representation of learning objectives (in section II-A we will see why), nor is it a complete representation of study material and teaching methods, but it is a subject-specific knowledge model of the course.

The broad structure of the whole methodology is shown in figure 1. This paper we will focus on the modeling process that is based on learning objectives and, though not strictly part of the knowledge model, briefly touch upon templates. See Schmidt et al. [3] for more in-depth information on the exercise generation methodology itself.

The modeling of courses is structured into four main steps and an optional templating step, see fig. 2.

Instead of looking at the course material first, we wanted to take a step back and started with the fundamentals of teaching, the learning objectives (LOs). We begin the modeling process by extracting, defining and redefining learning objectives (1), see section II-A. Afterwards the learning objectives must be categorized into a taxonomy (2). This gives an insight into the structure and difficulty of the course. Section II-C will discuss this step in more detail. The next step (3) is to create formally defined subject matters, called FDLs (German: **F**ormal **D**efinierte **L**erninhalte). These can be definitions, formulas, diagrams, arguments, etc. taken from course material. They

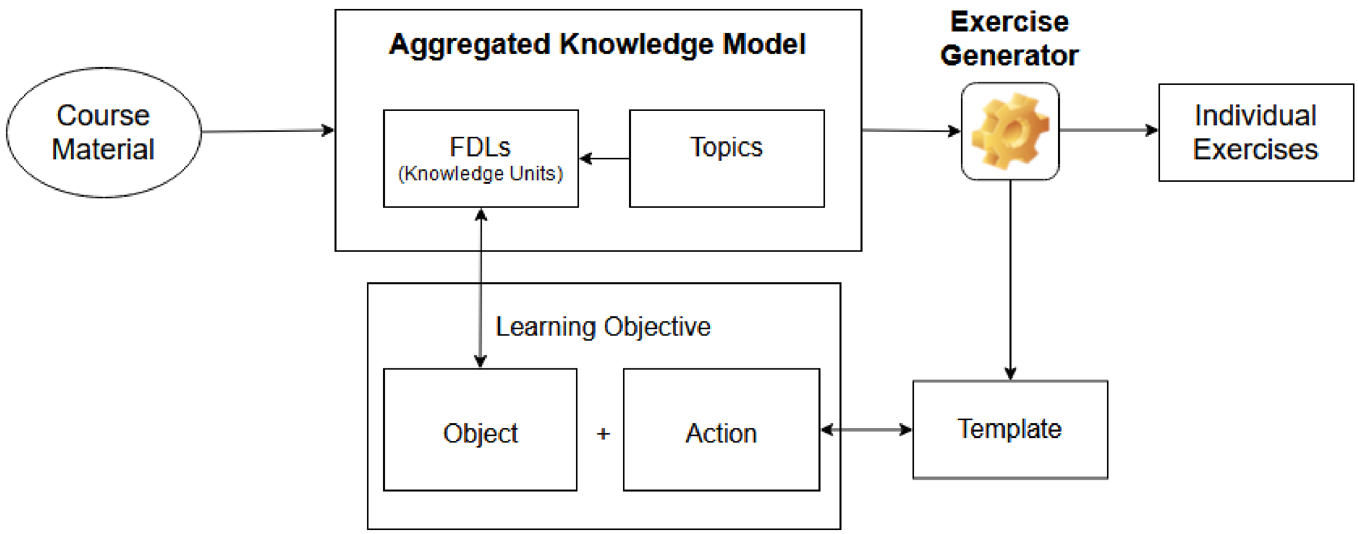

Fig. 1: Structure of the exercise generation: From the modeling of an aggregated knowledge representation to the creation of individual exercises.

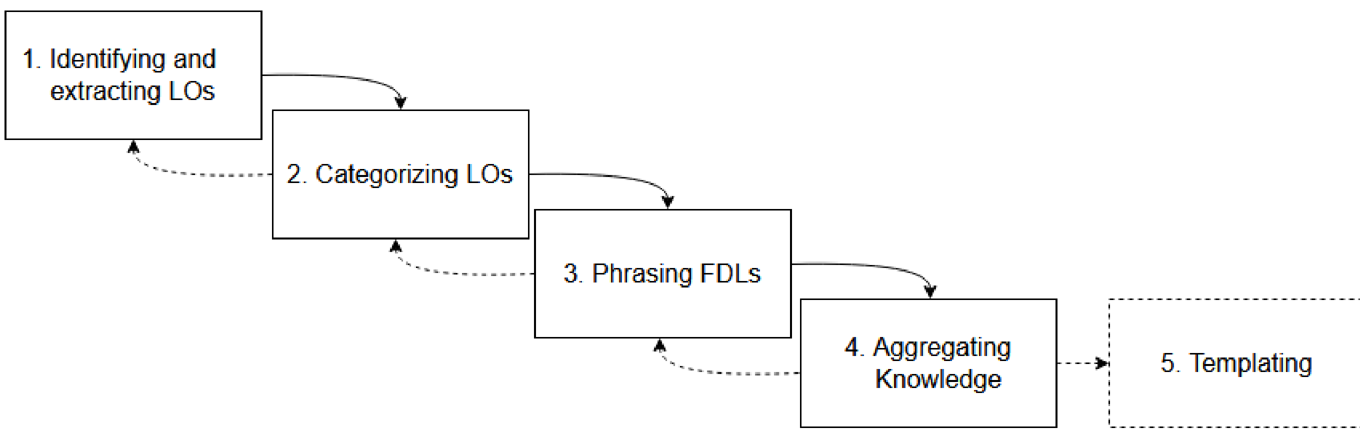

Fig. 2: Modeling steps: First learning objectives (LOs) are identified and extracted from course material (1). Afterward these LOs are categorized with the help of a taxonomy (2). Then Formally Defined Subject Matter (FDLs) are defined (3). Next FDLs are topical structured and combined in the aggregated knowledge model (4). Lastly, missing templates can be added (5). If required, previous steps can be amended.

can be thought of as digital, atomic knowledge units. Later the FDLs will represent the object in our learning objectives (see section III-A). In step (4) the FDLs are hierarchically structured via topics, thus forming the aggregated model. This representation is then mapped to corresponding learning objectives. By doing so we make sure that A) all required FDLs are included and B) no unnecessary FDLs are incorporated into the model. This has the advantage that teachers can see what knowledge units they actually need in order to fulfill learning objectives. Though the templating step (5) lies somewhat outside the scope of mere modeling it is still essential for the purpose of generating exercises and creating digital assessments of learning objectives. Templates are the semantic "husks" of tasks students should perform. A template takes a FDL of a certain type (e.g. characteristics) as input and produces an exercise in a desired format (HTML or tex / PDF), according to the template's implementation (e.g. 'select all characteristics of X from following list'). In our case, the templates already exist in our generator (and therefore not strictly part of the course modeling process). In other cases though, they must be extracted from existing exercises and / or exams, or must be newly designed. See section III-B for more details.

As development processes are often required to be flexible, previous steps can be returned to if needed. Such a scenario could be identifying a small niche topic in the aggregated knowledge model, which should then be brought closer to existing topics. This requires the redefinition of learning objectives and possibly changes in the FDLs and aggregated model. As a result, the overall process becomes iterative.

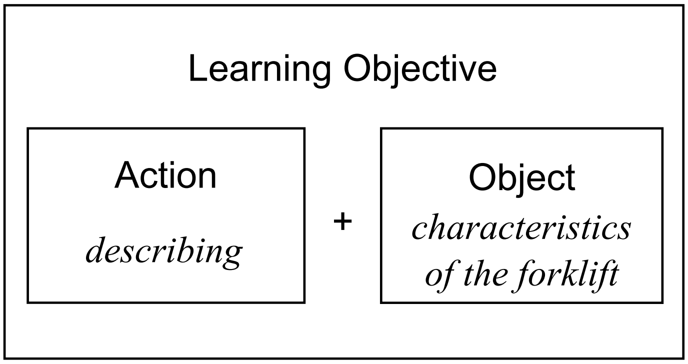

Fig. 3: Structure of a learning objective: An action (observable task) together with an object (subject matter)

II. Course Breakdown into Learning Objectives

Learning objectives are used to define the skills and knowledge students should have at a certain stage in a program [4]. Much has been said on the topic of outcome-based education since it emerged in the 1950s [5] and how to to formulate effective learning objectives [6]. We want to give a brief guide on the phrasing and classification process of LOs.

A. Defining and Phrasing Learning Objectives

It is important to note that learning objectives can be short term, long term, granular, or far reaching in scope. For our purpose we focused on the skills and knowledge students should have at the end of a semester and chose a high degree of resolution, meaning we phrase very specific LOs. One example in the realm of engineering education, specifically logistics, could be:

Students describe characteristics of the forklift.

This example contains the typical parts of a learning objective adapted after Schwier [7]: The audience (the students) and the behavior (describing characteristics of a forklift). Usually a external condition (describing in under 1 minute) and a degree (describe 20 characteristics) can be added. For our purposes a more practical and simplified construction is of value. Each learning objective consists of an action (describing) and an object (characteristics of a forklift) on which the audience acts upon, see fig. 3.

We want to prevent the use of modal verbs like "can" or "able", as well as unobservable behavior like "knowing" or "thinking". In the first case we do not want students to be theoretically able to do something, we want them to observably *do*. The latter is important for the evaluations of learning objectives. We cannot not know if students have "thought" about a problem or "know" about something because these are cognitive and mental phenomena that cannot be observed and therefore not be evaluated easily. Opposed to this, what we can observe (in tutoring or in exams), is if students actually calculate something or characterize the forklift.

Later in section IV-B we will map atomic knowledge units (FDLs) onto the object part of learning objectives. Templates of exercises will represent the action part of LOs, thus fully representing them in a digital form.

B. Extraction of Learning Objectives from Course Material

One way to extract learning objectives is to look at the study material and break it down into the units students should learn about (object) and how they are to do it (action). These materials can be the existing exercises, lectures notes, lab instructions, code, and so on. Another crucial factor should be exams. Ideally an exam should test if learning objectives are fulfilled, given that students had the chance to achieve these LOs beforehand. Learning objectives should be extracted and listed individually for each part of the course so the course structure can be evaluated in the sense of constructive alignment (see section II-D).

C. A Practical Learning Objective Taxonomy

Once learning objectives are found and phrased it is useful to categorize them in a taxonomy. This way the course content can be mapped onto different cognitive domains. The advantage in this is, that teachers can see the underlying structure of what they want to teach and how complex the tasks (actions) are they want students to perform. It also makes assessing courses and comparing them to other courses easier and lets the teacher keep track of students' growth [8]. Usually learning objective taxonomies describe some form of progression from less learning intensive tasks to more challenging tasks.

For the purpose of generating exercises, a taxonomy has another important benefit: It already gives a dimension of difficulty, more specifically, the complexity after Robinson [9]. This is the teacher-given level of difficulty and may vary from the difficulty the students perceive. Depending on the taxonomy used, *describing* might be a relatively simple action to perform. On the other side, *analyzing* is usually a complex task that will occur on a higher level of a taxonomy. When an item based on a learning objective with the *describing*-action is created we know that it belongs in a certain range of difficulty. Of course, this does not encompass all dimensions of difficulty. Two descriptions might still vary in difficulty depending on the object and conditions regarding the question design (single choice, multiple choice, essay etc.). But it gives a first parameter to adjust difficulty in our system. This plays an important role in the creation of individual exercises.

The most well-known learning objective taxonomy comes from Benjamin Bloom [10]. In this taxonomy the learning objectives are divided into six levels in order of increasing task complexity: knowledge, comprehension, application, analysis, synthesis and evaluation. Later Krathwohl [11] built upon this taxonomy and modernized it. Even though Bloom's work is influential it is not without criticism [12]. With its six domains it can be difficult to discern the fine lines between the domains (*Is this exercise more in application or synthesis?*). In our experience, teachers had some problems deciding between two or even three domains, especially in the higher tiers. It might also not be obvious why *comprehension* is less complex in terms of cognition than *analysis*. This confusion takes up time in the process of defining learning objectives.

For this reason we decided to use a taxonomy with only three domains: Knowing, Applying and Understanding. It is freely based on the works of H. Mandl[1] and supports a more intuitive and less specific way to look at learning objectives. Besides its three domains, the used taxonomy has two dimensions: Content-Type-Dimension (Subject-specific and Methodical) and Task-Dimension (Knowing, Applying and Understanding), see fig. 4. We call it KKV-Taxonomy (German: **K**ennen **K**önnen **V**erstehen [Knowing, Applying, Understanding]).

In the following the dimensions of this taxonomy are briefly explained:

1) Content-Type-Dimension: Subject-specific and Methodical: - The row dimensions of the taxonomy are concerned with the type of educational content. We chose to represent subject-specific content and methodical content, though there could be more, e.g. social or personal, but we opted to exclude these because our exercise generator cannot support social or personal learning objectives and we wanted to keep the taxonomy as lean and practical as possible.

Subject-specific content relates to the topics discussed in a given course. This could be ship engines or Newton's third law. On the other hand, methodical contents are the techniques students should develop and use when working with subject matter. It is to note that the methodical content of one course could be subject-specific content in another. In a physics course reducing fractions might be a necessary technique (methodical content) to solve a problem concerning thermodynamics (subject-specific content), but in a math course reducing fractions is the subject matter. For every course these need to be evaluated individually.

2) Task-Dimension: Knowing, Applying and Understanding: - In the task-dimension three depths of cognitive tasks are described. Usually these ascend in difficulty with *knowing* being the easiest and *understanding* the most difficult. *Knowing* concerns the recalling of objects. This could be citing Newton's third law (subject-specific knowing) or describing the method to reduce fractions (methodical knowing). Observable actions for learning objectives could be *state, name, define, label, recite, list,* etc.

One step higher students should not only memorize knowledge, but also *apply* it in some way. Usually knowing some facts and methods is required here, so a natural progression is given by this taxonomy. Solving a problem with Newton's third law or reducing fractions might appear. Common actions are: *calculate, compare, complete, illustrate, translate,* etc.

The highest and most demanding level is *understanding*. Here students should be familiar with complex circumstances and handle challenging problems. Creating something new based on prior knowledge and experience, as well as analyzing and evaluating objects show a deeper understanding. Students might need to design an experiment showcasing Newton's third law or find a rule to reducing compound fractions. Learning objectives in this category could contain: *evaluate, justify, design, synthesize, create,* etc.

D. Evaluating and Analyzing Courses

After extracting and categorizing learning objective they can be used to inspect the didactic soundness of the originating course.

By creating a separate list of LOs for each source, teachers can observe the degree of constructive alignment. This is the idea that the taught content in each aspect of teaching, so for example the lecture, the tutorials and the exam, is congruent [13]. Since we extracted the LOs and categorized them individually, we can evaluate if they align, meaning that we should find similar learning objectives in all parts of the course or that they build upon each other. For example, a lecture might cover certain facts and present examples (*Knowing*). In a tutoring session these are then practiced by the students (*Applying* and *Understanding*). Of course these abilities could and should be tested in an exam. This inspection of a course and its constructive alignment may lead to the creation of new learning objectives, the exclusion of existing ones or even the restructuring of a course. One might find some fundamentals in the knowing column to be missing, resulting in the failure to achieve learning objectives in higher difficulties, or that some easier learning objective do not lead to anything more profound later on.

As mentioned previously, dimensions of our taxonomy also imply difficulty. Thus, we can observe how many of our learning objectives fall into the *knowing*, *applying* or *understanding* column. This reveals information about the course structure and its difficulty, creating an opportunity to reflect upon teaching methods, setting, and the prerequisites students should have beforehand.

III. Incorporation into an Exercise System

The following sections address how the previously described theories can be integrated into a practical system. For this we implemented the 'GENIE' exercise system, which works using our new exercise generation methodology, as described in [3]. To summarize, this new exercise generation approach is based on formally defining knowledge and creating models of it, using a course's subject matter. The generator is then able to generate exercises from said models with the help of various templates.

A. Formally Defined Subject Matter (FDLs)

Formally defined subject matter (German: **F**ormal **D**efinierte **L**erninhalte) is the first of two key aspects of our digital learning objective definition. As also described in more detail in [3], there exist different types of knowledge which are formalized using their corresponding FDL type. In order to capture all the subject matter present in our two tested courses, we identified the following types of knowledge: definitions, examples, characteristics, trade-offs, sequences, (labeled) figures,

[1] Unfortunately, to our knowledge Prof. Dr. (Emeritus) Heinz Mandl has not published these ideas, but instead utilized them in his lectures, while also passing them on orally.

	Knowing	Applying	Understanding
Subject specific	*Concerning facts*	*Task related*	*Evaluation and creation*
Methodical	*Knowing method*	*Applying method*	*Extending method*

Fig. 4: Learning objective taxonomy based freely on H. Mandl.

equations, constants, patterns, hierarchies and progressions (continued measurements). Each of these types' key aspects are captured by the corresponding FDL type as a new model, resulting in a repository of "atomic" knowledge units.

As an exemplary scenario, the lecturer wishes to teach about different aspects of conveyor systems. Each of these aspects is saved in a fitting FDL. One of said aspects is supposed to be learning about some specific conveyor systems to get a better understanding of the more abstract and general idea of conveyor systems. Figure 5 depicts a possible example-type FDL containing information on such concrete conveyor systems. As the examples' terms are not sufficient, some examples also include references to images depicting them. Lastly, conveyor systems can be categorized by certain attributes.

B. Templates

Templates are provided by the generator and define how the subject matter from a specific FDL-type is transformed into an exercise of a certain type. Given the previous example FDL in figure 5, one template could use example-type FDLs to create multiple-choice exercises by selecting some of the given examples, as well as a number of distractors from other example FDLs, and asking the student to select all examples of "conveyor systems". A second template could ask the student to name n conveyor systems, accepting any solutions which are defined in the FDL. Both of these templates operate on the first KKV-taxonomy level of *knowing*, but the form of the solutions differ between multiple-choice and textual input. This brings forth two major advantages:

1) A specific template can be used to achieve a concise coverage of a learning objective.
2) Different exercise types within the same taxonomy level, stemming from the same subject matter, are similar enough to be close to interchangeable. This allows for more exercise variety and possibly better learning by taking minorly different views on the same object.

Aside from FDL(s), templates also take a list of parameters as input. These can be used to further individualize the resulting exercise, e.g. by making the number of correct and wrong options in a multiple-choice exercise dependent on the difficulty parameter (1: few options vs. 10: many options).

IV. Aggregated Knowledge Model Creation

In the last step FDLs are topically structured into a network, thus forming the aggregated knowledge representation.

Having created a FDL repository from the course materials there still is a lack of structure to it. Courses are often structured on a topic basis. Our two courses are each split into four larger sections and then further into smaller topics with subject matter (FDLs) possibly contained within each topic. We deemed sorting the FDLs after a course-representative structure to be the most sensible approach in order to attain a by human and machine comprehensible, overarching model.

With an existing course being taken as a basis, a topical structure already exists, meaning LOs and FDLs can easily be sorted into it, so that every LO is connected to one, or in many cases several, FDL(s). Any leftover FDLs need to be critically evaluated, since at this stage they don´t seem to be necessary to fulfil learning objectives. In this case more LOs might be created, indicating missing coverage or the unconnected FDLs are simply discarded to keep the course as lean as possible.

In theory the knowledge model can also be created using only LOs, as would be the case when creating a new course from scratch. This would change the order of steps in the process, but ultimately deliver the same result after further creating course materials, extracting FDLs and linking them to topics.

A. Course Improvement Potential

When creating an aggregated knowledge model, one is confronted with two questions which do not have clear answers, but at the same time the possible solutions can be weighed against one another. Solutions following certain guidelines

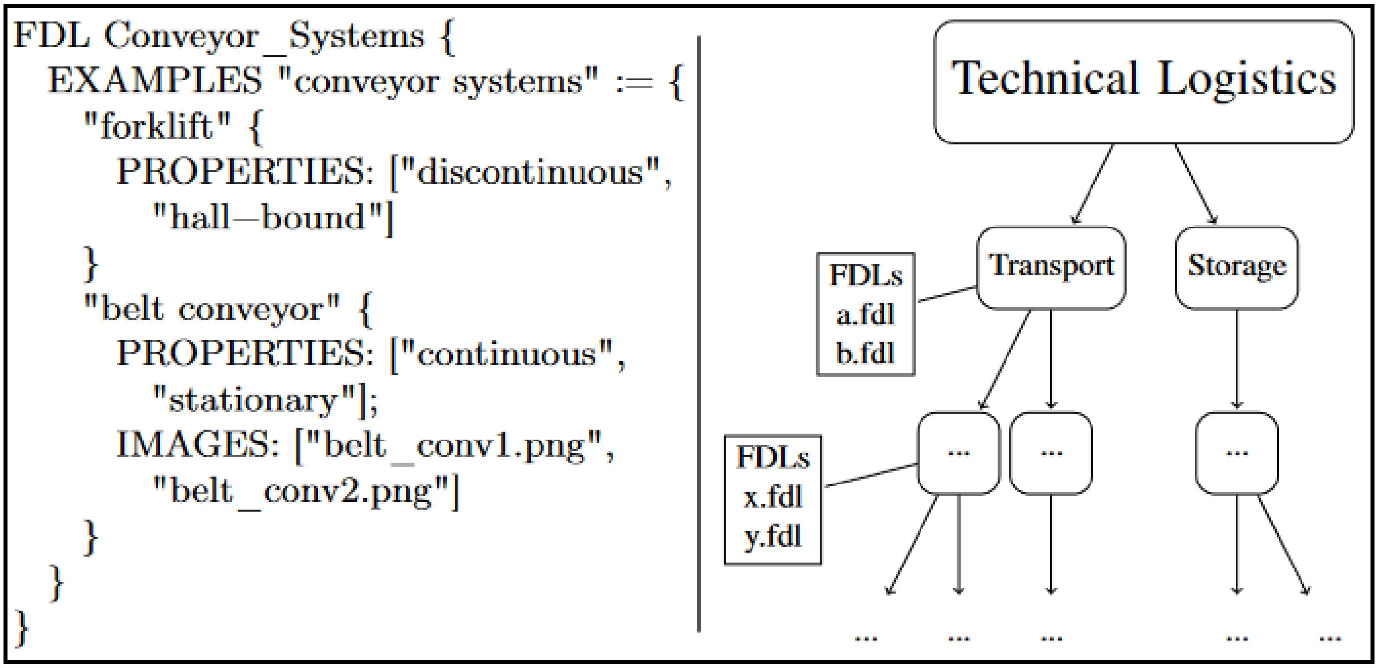

Fig. 5: Left: An example-type FDL defining instances of conveyor systems with attributes and images linked to them. Right: Exemplary aggregated knowledge model consisting of topic nodes containing FDLs, ordered in a tree structure.

should have a higher quality and thus be preferred. Both questions will be discussed in turns:

1. What are the (relevant) topics?

Given a set of learning objectives and linked FDLs, devised from an existing course, reconstructing the topic structure is fairly simple. The resulting tree can then be critically analyzed by looking for specific occurrences:

- Topic nodes without FDLs - While it can be advantageous to insert empty topic nodes as a way to better organize subtopics, too many empty topics following one another may imply that the following subject matter is not properly introduced or in which environment it lies is unclear. Furthermore, leaf (childless) topic nodes should never be empty.
- Topical outliers - Long topic strings, which do not branch out, but instead stay as a singular "strand", lead to the case of the final topic being very niche and unrelated to other contents of the course. It is worthwhile considering removing such topics or fleshing them out with closely related contents.

In summary, identifying relevant and "well-fitting" topics allows for a cleanly compressed and enclosed course.

2. In what order should the topics be structured?

The second challenge is to sort the identified topics, either vertically (parent-child relation), or horizontally (sibling relation). Lecturers usually are already able to fulfill this in a didactically sound way using their intuition or common sense. The relation between topics being defined the way it is ($A \searrow B$: topic B is addressed within the topic of A) already leads to the subject matter's environment being touched upon before being introduced / defined itself. It should be noted that the order of the model does not necessarily represent the order in which the contents are taught. Introductory examples for more general concepts is a commonly used teaching strategy. Nevertheless, a few points are worth pointing out:

- Vertical order - A tree structure implies that nodes (except the root) only have one parent. This forces more general topics to be a parent of their subtopics, which are interchangeable with abstract and more specific concepts or general and more concrete objects.
- Horizontal order - As a convention we say that topics are taught depth-first and left to right. This implies that sibling nodes are best ordered depending on which subtrees can help in or are necessary in understanding the others. If no such dependencies exist it can also be considered ordering them by size or difficulty of the contents.

Concluding, the tree structure already forces teachers to design the aggregated knowledge model in an intuitively "correct" way. It leaves them with deciding on the relevancy of certain topics and in which order they wish to teach the contents.

B. Connecting Learning Objectives, FDLs and Templates

Reaching this step there should exist a set of categorized learning objectives, a set of FDLs and an aggregated knowledge model, although the latter is strictly speaking not required here. The templates needed to finish the digitization are provided by the exercise generator. As shown in fig. 1, each learning objective now is assigned one or more FDL(s) and one template. Should two or more learning objectives share the same FDLs and template with another we consider them to be duplicates or similar enough to function as duplicates, allowing for the duplicates' removal. This guarantees a set of truly unique learning objectives.

Given how the exercise generator exactly takes a template and FDLs as input to generate an exercise, we can assure that the generated exercise is truly an assessment of the learning objective connected to them, making the assessment correct-by-design.

V. Practical Example

Though the focus of this paper is to present the theoretical substructure to generate exercises we still like to showcase a concrete example of a practical application. The following example illustrates how a FDL and templates are used to generate three different types of exercises which fulfill three different learning objectives, see fig. 6.

As a reminder, the general idea of our concept is that if a FDL is aligned with the LOs of a course, the exercises generated using specific templates inherently target said LOs.

As an example, the course 'Technical Logistics' includes various LOs concerning calculations related to forklifts, specifically:

(a) Students calculate the engine power of forklifts.
(b) Students name the relevant variables for calculating the engine power of forklifts.
(c) Students deduct the units of variables required to calculate the engine power of forklifts.

These LOs are all encompassed by an equation-type FDL concerning the formula $P = \frac{F \cdot v}{\mu}$, where the variables represent the engine power P, the force required to overcome the rolling resistance F, the maximum velocity of the forklift v, and the efficiency of the engine μ.

The generator now provides different templates compatible with equation-type FDLs. Here there exist respective templates producing exercises targeting said LOs, in particular:

- Standard calculation templates roll random values for all except one variable and task the student to find the missing value. Here both selecting the correct value, as well as inputting it fulfill the LO (a).
- Naming variables templates task the student with selecting present variables from a multiple choice list, fulfilling LO (b).
- Templates for deducing units can be seen as equivalent to normal calculations, except for values being left out. As such the student must 'calculate' the unit of a selected variable, given the units of the remaining variables, thus fulfilling LO (c).

In general this means that exercises produced by the generator are, by design, appropriate assessments of certain types of LOs, given that these are aligned with a FDL.

VI. Evaluation and Conclusion

Our goals were to create a representation of a course through a modeling process that stimulates didactic reflection without losing its technical usability (machine-readability). The advantages, but also the limits of our modeling process are discussed in the following.

A. Advantages

With a heavy focus on learning objectives and constructive alignment, this modeling method gives teacher a chance to evaluate the materials they want to teach and realign their lectures, tutorials and exams. While this is something the faculty should do from time to time or when creating new courses anyways, we combine these didactic examinations with the digitization of courses, two important aspects of modern education, into one work flow.

Another advantage is that the final model stays lean (if handled correctly), since before adding new content in form of FDLs one should first create new or find preexisting learning objectives. When not done this way the system can identify "loose" FDLs, that serve no learning object, and therefore are questionable. Through the use of this model teachers should always think of the knowledge units and skills students need in order to reach certain learning outcomes.

Furthermore, combining the model with our exercise generator allows for the automated assessment of the defined learning objectives using a high variety of possible exercises. The students' skill / progress can be easily tracked at different levels, e.g. per topic. This opens the possibly for intelligent tutoring systems and learning evaluation that is closely connected to the learning objectives.

Lastly, our approach is not bound to a certain field. While we applied it to engineering sciences, it can be adapted to fit any other subject.

B. Limits

Although the translation of definitions, formulas and graphics into FDLs works smoothly, other knowledge types are harder to abstract. In general, we found that formalizing knowledge for learning objectives on the *Understanding* level to be much harder. Actions like *interpret* or *analyze* usually need a greater number of complex FDLs to truly represent the object of the learning objective. Considering the large number of learning objectives, the implementation can be very time consuming or not yet possible. Specifically procedural knowledge has not been formalized yet, making use of methodical learning objectives difficult.

Additionally, this model was explicitly created with the generation of exercises in mind. This purpose motivated the design and might not fit every other digital outcome.

On the didactic side it should be noted that learning objectives as the corner stone of outcome-based learning are not without criticism. This way of thinking about education and learning has been called narrow and restrictive [14] and lacking ways for students to develop their creativity [15]. Learning objectives do not solve all didactic problems and can only be a part of effective teaching and learning. When deploying the presented modeling process this should be kept in mind.

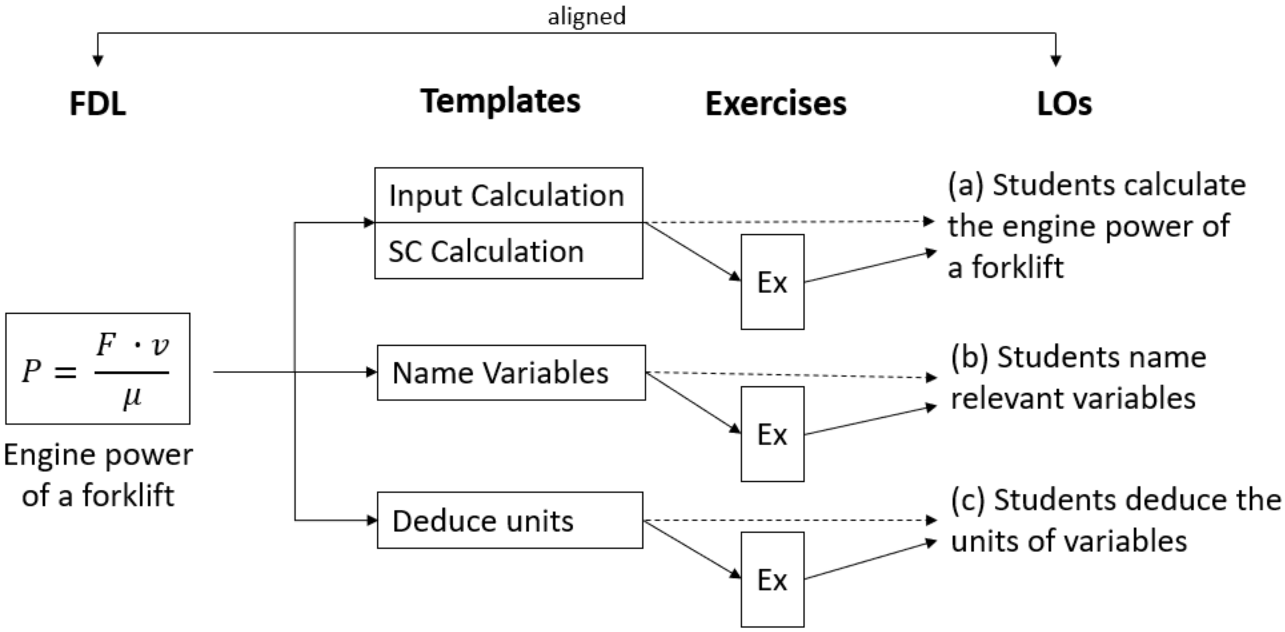

Fig. 6: Aligned structure of a FDL, possible templates, the resulting exercises and the underlying learning objectives.

C. Conclusion and Future Work

In this paper we presented an applied modeling process of course material into an aggregated knowledge representation for the use in automated exercise generation. The modeling process is based on learning objectives to help with the constructive alignment, difficulty, and content assessment of modeled courses. The digitization of the learning objectives through formally defined subject matter (FDLs) and templates was discussed.

Course digitization as presented here enables processes based on machine-readability and standardization. Such processes could include advanced course analysis and comparison, course/knowledge preservation, or even intelligent tutoring systems. In the future we want to incorporate ways for teachers to interact with an already existing model and add content (LOs, FDLs and templates) easily through user-friendly tools, thus cutting down on the time investment. Furthermore we want to evaluate the teacher experience when working with our modeling method and investigate the didactic advantages and disadvantages.

Financial Disclosure

The research project "GENERATING" is funded by grants provided by the *German Federal Ministry for Education and Research* (BMBF).

References

[1] Aleksandrov, A., Tsvetkov, Y., Zhileykin, M. , *Engineering Education: Key Features of the Digital Transformation*, ITM Web of Conferences. 35. , 2022

[2] Abumandour, *Applying e-learning system for engineering education – challenges and obstacles*, Journal of Research in Innovative Teaching & Learning, 15, 2, 2022

[3] D. Schmidt, M. Uzulis, N. Kersten, O. Grasse, J. Hinckeldeyn and J. Kreutzfeldt, "Foundations for Complete Exercise Generation," 2022 20th International Conference on Information Technology Based Higher Education and Training (ITHET), Antalya, Turkey, 2022, pp. 1-7, doi: 10.1109/ITHET56107.2022.10031913

[4] Melton, R. *Learning Outcomes for Higher Education: Some Key Issues.* British Journal of Educational Studies, 44(4), 409-425, 1996

[5] Tyler, R. W., *Basic Principles of Curriculum and Instruction*, Illinois: Chicago University Press. 1950

[6] Ferguson, L., *Writing learning objectives*, Journal of nursing staff development : JNSD, 1998

[7] Schwier, R.A., *Objectives for Instruction*, 1990

[8] Briggs, D.D.,Diaz-Bilello, E., *Using a Learning Progression Framework to Assess and Evaluate Student Growth* , 2015

[9] Robinson, P., *Task Complexity,Task Difficulty and Task Production: Exploring Interactions in a Componential Framework*, 2001

[10] Bloom B. S. , Engelhart M. D., *Taxonomy of Educational Objectives*, Handbook I: Cognitive Domain, 1956

[11] Krathwohl, D.R., *A Revision of Bloom's Taxonomy: An Overview*, Theory Into Practice, 41:4, 212-218, 2002

[12] Colder, J. R. "In the Cells of the 'Bloom Taxonomy '." Journal of Curriculum Studies 15.3 (1983): 291-302.

[13] Biggs, J., *Enhancing teaching through constructive alignment*, Higher Education 32, 1996

[14] Schwartz, G., Cavener, L. A. *Outcome based education and curriculum change: advocacy, practice and critique*, Journal of Curriculum and Supervision 9(4): 326-8 1994

[15] Towers, J. M. , *The perils of outcome based teacher education*, Phi Delta Kappan, 1994