A Knowledge-Model for AI-Driven Tutoring Systems

Andreas Baumgart[a,1] and Amir Madany Mamlouk[b]

[a] *Engineering and Computer Science, HAW Hamburg, Germany*
[b] *Neuro- and Bioinformatics, University of Lübeck, Germany*

Abstract. A powerful new complement to traditional synchronous teaching is emerging: intelligent tutoring systems. The narrative: A learner interacts with a digital agent. The agent reviews, selects and proposes individually tailored educational resources and processes – i.e. a meaningful succession of instructions, tests or groupwork. The aim is to make personal tutored learning the new norm in higher education – especially in groups with heterogeneous educational backgrounds. The challenge: Today, there are no suitable data that allow computer-agents to learn how to take reasonable decisions. Available educational resources cannot be addressed by a computer logic because up to now they have not been tagged with machine-readable information at all or these have not been provided uniformly. And what's worse: there are no agreed conceptual and structured models of what we understand by „learning", how this model-to-be could be implemented in a computer algorithm and what those explicit decisions are that a tutoring system could take. So, a prerequisite for any future digital agent is to have a structured, computer-accessible model of "knowledge". This model is required to qualify and quantify individual learning, to allow the association of resources as learning objects and to provide a base to operationalize learning for AI-based agents. We will suggest a conceptual model of "knowledge" based on a variant of Bloom's taxonomy, transfer this concept of cognitive learning objectives into an ontology and describe an implementation into a web-based database application. The approach has been employed to model the basics of abstract knowledge in engineering mechanics at university-level. This paper addresses interdisciplinary aspects ranging from a teaching methodology, the taxonomy of knowledge in cognitive science, over a database-application for ontologies to an implementation of this model in a Grails service. We aim to deliver this web-based ontology, its user-interfaces and APIs into a research network that qualifies AI-based agents for competence-based tutoring.

Keywords. Competence-Based Learning, Knowledge Concept, Ontology, Web-Service, Taxonomy of Knowledge, Web-Application

1. Introduction

Digital learning is seen as a new and promising approach in creating a more effective and efficient teaching environment, in delivering higher returns on education. One line of development has been to provide structural elements like educational resources and courses online – many of them being freely accessible or openly licensed as in HOOU [17] – or to provide environments for testing domain-specific knowledge – as in MINTfit [22].

---

[1] Corresponding Author.

The other line of development addresses processes required to implement educational resources into active learning, including pedagogies as e.g. blended/hybrid learning or gamification. A distinguishing feature between these processes is the level of individuality that a teaching environment can offer – from addressing the full group of students with the same interventions to addressing one individual student with specific, tailor-made interventions. It is assumed that an individual offer to a student creates improved learning opportunities in a significant number of settings (Wood [30]). Such an individual approach could be incorporated by a tutoring agent, an intelligent tutoring system (Graesser [12]).

This agent must – in order to add value to any teaching environment – have a conceptual understanding of the teaching-subjects and their dependencies. And such a conceptual model in computer accessible form does not exist yet.

A prototype for such a model is being presented in this manuscript.

2. Structures and Taxonomies for Knowledge

With knowledge being a substantial competitive factor for individuals and national economies alike, scientists have long developed models to make learning-outcomes measurable and comparable. This section describes, how existing concepts have been integrated into our approach.

2.1. Learning Analytics

Learning Analytics and its scientific research field Educational Data Mining describe ways to employ data - e.g. test-results - generated by students when using e-learning platforms. These data can be used – subject to data protection regulations - to monitor learning behavior and progress of an individual or a group of students, the data can be analyzed with statistical methods, visualized and can be employed to compare between groups, between teaching methodologies employed for learning and between individual teacher personalities or could drive computer adaptive testing. However, these data are usually not appropriate for an AI approach to learning:

Data acquired by Learning Management Systems (LMS) are mostly points that students achieve in tests. These tests usually have no metrology - no measurement system - for knowledge in place. A common example is a succession of tests (quizzes) in a course implemented in Moodle. Each test delivers an aggregated point-value which stands implicitly for a certain achieved learning outcome. The point-values have no meaning beyond the test.

As a remedy, the LMS "Moodle" offers Competency Frameworks. These are hierarchical tree structures with a parent / child -structure. So Mechanical Engineering could have "Solid-State Mechanics" and "Fluid-Mechanics" as children, "Solid-State Mechanics" could be the parent of "Homogeneous Materials" and "Heterogeneous Materials". The test-performance in child-concepts can then be employed to indicate performance in parent-concepts. This approach creates structure – a semantic tree - but not a meaningful context between knowledge items. The tree might assign "Vector-Algebra" as a child-node of "Geometry" in the Competency Framework "Mathematics" – but it is also the pre-requisite for understanding equilibrium conditions in statics – which is engineering or physics.

As a remedy, Halloun [13] designed tests some 40 years ago to measure students' understanding in the application of physical concepts, e.g. "force". Questions in these tests are not confined to a singular knowledge-item like "force" but aim to find indicators for the degree of conceptual-understanding that students have.

All questions in such a Concept Assessment Test (CAT) are subject to a rigorous scrutiny, they are therefore called evidence- and research-based questions. To give reproducible and meaningful results, CATs must follow a strict standardized approach: no question must be altered or replaced, no questions must be added and the time available to solve all questions is fixed. In engineering, the CAT for statics, the "CATS", consists of 27 questions for students in the first semester of engineering mechanics. It tests for 8 concept-categories with "static equilibrium conditions" being one of them. So, CATS can be used to measure mastery of a set of learning concepts from engineering e.g. after a course when comparing different teaching approaches as has been described successfully by Direnga [10] or with a similar intent by Adams [1].

Another approach is the item response theory (IRT) which comes with a dedicated statistical analysis approach to the problem (Hambleton [14]). In contrast to CATs, questions can be designed as needed and can have different weighing factors when contributing to the overall result.

Common for both approaches though is that the concepts or items tested for have no explicit meaningful connection to each other – their interrelation can only be established by a discussion between the designers of the tests. And both approaches have no objectively defined level of reference-knowledge defined – the skill-level achieved when obtaining all points in a test is not

Example: As the author of a test, I will have a subjective understanding of the conceptual knowledge that students have attained, and the points not awarded indicate the distance to the intended learning objective. I could pass on this test to a colleague and explain my understanding of the test and the targeted learning objectives.

For an AI, this approach is "incomprehensible", it would need more structure, a schema of knowledge to adhere to. Knowledge must therefore be decomposed into distinct and explicit subjects of suitable complexity – as in the concept-categories of CAT. But these concepts should not be confined to be stand-alone items in one individual course at one university. Example: "static equilibrium conditions" is not a sufficiently distinct concept as it may refer to an approach with force equilibriums or with analytical methods as in the Principle of Virtual Work.

And the structure should incorporate an objective skill-level that indicates the learner's performance in employing the subject to solve a problem. Example: a student may compute the forces required to achieve "static equilibrium conditions" in a practice-and-drill approach or analyze geometric conditions under which no solution to the problem exists, with the latter being significantly more demanding.

2.2. A Schema for Learning Objects

Similar to the approach of structuring knowledge is the idea to structure learning objects to an agreed schema. In IEEE [19] a schema is proposed for the representation of interoperability conditions of learning objects using learning objects metadata (LOM). This standard uses XML Schema definition to describe and define an aggregation relationship between data items to considerable detail: it specifies e.g. roles, resource types and interactivity levels of LOM. Like Moodle Competency Frameworks, it creates a tree-structure of parent/child elements. But in this approach, a child-element can have

a specific composition attribute, e.g. can be a "Requirement" or an "OrComposite". And it adds difficulty levels – from "very easy" to "very difficult".

2.3. Consolidated Approach

Both Moodle and IEEE approaches add a text-book-like structure to knowledge-elements and LOM. Additionally, IEEE offers a rough taxonomy (from "easy" to "difficult") e.g. for an exercise. They both provide a qualitative structure of a body of knowledge.

Krathwohl [4] creates a detailed taxonomy of knowledge for one individual knowledge-base aimed at quantifying skills. But its concept includes no relations between the knowledge-bases and was not designed to be implemented in the context of information technology.

Instruments like Concept Inventories and IRT describe in considerable technical detail a confined element of knowledge to be tested for and focus on the analytics to assess a learners' skills. The target of the analysis is a statement as "concept understood / not understood" and does not include a rationale for the quantitative manifestation of knowledge so it could be "read" by a computer.

And none of the above account for the semantic relationship between learning objects.

In Chi [8] and Paquet [23], the focus is tuned to the modelling of semantic networks for knowledge, the latter targeting computer applications. Paquet's work provides an excellent overview and describes the generic approach to engineering ontologies and the abstract objects to be employed. However, this approach does not explicitly incorporate Krathwohl's taxonomy (though it refers to Bloom's taxonomy) and the models are not built as an interactive Web-application to be used by tutoring agents.

What we aim for is a composition of key features from these references. Our representation of knowledge must be tailored to provide structured access to human knowledge for a computer application so it can mimic decisions in tutoring scenarios. Our approach is to

- breakdown a knowledge domain into discrete items,
- represent inferences between items that allows for an abstraction of dependencies between items,
- include an explicit continuous taxonomy that quantifies for each knowledge item a student's performance or skill to solve specific problems,
- describe this taxonomy for discrete, uniform, hierarchical taxonomy level prototypes – synonym to Krathwohl's categories – represented by one or several statements describing a learner's capability to perform a task and
- incorporate these items in a collaborative, web-accessible platform.

This approach to knowledge representation will not – as it may appear – degrade an agent to adhere to formal rules (Alven [2]) or simple "if-then-else" approaches. But in a situation where no structured data from students are available that represent a learning process, a tutoring algorithm can start teaching. It can derive learning tracks from the links in the knowledge representation, propose to either deepen the understanding in one knowledge-base or broaden the knowledge by adding new neighboring bases instead. This, however, is pure guesswork or a random approach. We do not think that this is a major drawback for a tutoring agent, since my strategy as teacher is similar – even though I may not be aware of the concrete decisions that I take in a specific scenario.

Without data being available on "success-tracks" in our learning landscape, an algorithm cannot decide, which decision is best – neither statistically nor individually. In other words: there is no Dijkstra- or shortest-path-algorithm for this problem. But any student consenting to having his or her data of the learning process to be recorded adds a meaningful set of data to the collection. Heuristics as "first deepen a subject, then broaden" may emerge and develop to providing meaningful, individually relevant teaching interventions at specific branching points. With each individual learning experience recorded and made available to other agents, an agent may see a "beaten" track of successful learning emerging. This will be the kick-off for a novel recommendation system. And this is where AI comes into play.

3. Concept of Knowledge

Our model of knowledge shall be an agreed, comprehensive and complete representation of abstract subjects to know. We use "mechanical engineering" as an example for a knowledge domain because it is well suited for modelling conceptual knowledge and has a consolidated, agreed body of knowledge. If the complete landscape of structured knowledge exists today in any form, it is locked in the minds of university professors and teachers. What we intend is to project this mind-model into something that finds its symbolic representation in a computer system and is sufficiently standardized so that different people would come up with similar knowledge-models.

So far, we have been using the catchy expression "knowledge" to name the intended outcome of learning. Knowledge, however, is not a standardized and not an unambiguous expression. To make things more complicated, we address different science domains – educational psychology, the modelling of mental structures, web-application frameworks, instructional methodology and engineering mechanics – which introduce their own subject-specific nomenclature. We will therefore "integrate" expressions from these domains and attempt to create a consolidated system of terms.

The connecting element in this approach shall be our vision of how AIs would employ the knowledge-model-to-be. As a teacher, I like the idea that our graduates learn to address professional problems in engineering by sketching and discussing a subject on a paper napkin over lunch. My picture implies that these engineers have a good command of a common technical language, need not to retreat for days of consultation to come up with a first justifiable opinion and understand the basics of their business, e.g. coordinates, principles, axioms (like the equilibrium of force) intuitively. A teaching approach which explicitly aims at these practical skills is "competency-based learning". Therefore "competence" shall be the lead-term in the development of this model.

Competency-Based Learning is characterized by properties, that we are addressing with a tutoring system in a similar way: its concept focuses on learning outcomes, the learner itself and a differentiation of appropriate learning approaches.

Foremost, learning outcomes are described as clearly observable performances of a student. This does not imply a particularly (high) level of mastery nor a point rating in terms of "8 out of 10", but rather an approach that describes concrete levels of achievements.

The student in this context is seen as an individual – having a unique personal, social and methodological background. Teaching should account for this background in the sense that any knowledge and any qualified perception of this background should be accounted for in teaching.

Finally, differentiation implies to account for this individual background in a most appropriate way, e.g. in the proposal of group- or single-work, the choice of learning objects or interventions as status-tests.

In this sense, competency-based learning phrases a concrete and realistic strategy for tutoring by agents using our knowledge-model. And it created an agenda of how to translate the above picture of engineers debating over lunch into a concrete definition of learning interventions.

As in the design of a curriculum, the learning objectives have two dimensions: the subjects like "static equilibrium of forces" and a depth of their understanding.

In our model for cognitive knowledge, we call a learning-subject

- a "competence-base",

their inferences

- a "competence-relation" and

the depth of understanding of this subject

- a competence-level.

Our competence-level-prototypes will describe distinct skill-levels and are identical to Krathwohl's categories. The new expression used intends to strengthen the consistency of terminology around the "competence"-concept.

3.1. The Knowledge Map

We approach this model with a hiking-analogy to learning: knowledge is seen as a landscape, our model is a map that represents the topology of the landscape and our student hikes in this landscape.
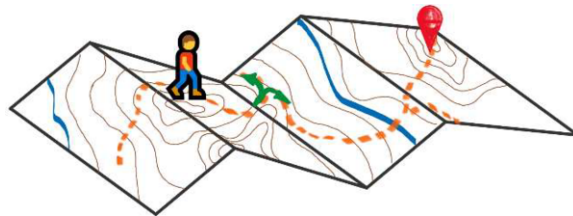

Figure 1. The Imaginative Learning Landscape.

An AI-agent would know from online-tests, "where" the student is, would know the topology of the map and the next target from the curriculum specification. Using information from the map, an AI would guide the student in this landscape employing waypoints and routes. So we first need to provide the map of a knowledge-landscape, including topology and infrastructure information.

Initial step to our model is a discretization and mapping of the learning-landscape to a network of competence-bases:
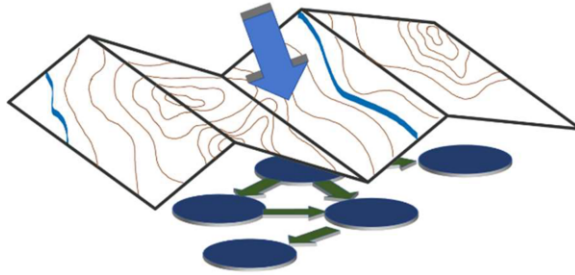
Figure 2. From the Map to a Discrete Network of Competence-Items and Relations.

Each competence-base (node) is connected to neighbors by relations (edges), these relations express directed conceptual inferences. The learning-items in a curriculum-design using our knowledge-model would thus be defined by selecting competence-bases. Then, the level of practical application skills shall be quantified as competence-levels.

3.2. A Semantic Network for Knowledge

This definition of learning-targets as competence-levels is distinct and unambiguous – if the knowledge-model is consistent. An agent can now follow all relations in the knowledge-model and identify all pre-requisites for achieving a learning target within a course and – by employing the conceptual inferences between the competence-bases - break it down to learning sessions, educational resources, and tests.

In this way, an agent will be able to assess the competence-level associated with a competence-base of an individual learner by appropriate tests. In our hiking-analogy, the agent queries all possible learning paths to the "GPS" coordinates of the learning objective. By weighing the current level of knowledge of a learner and the available learning material, the agent with AI can then optimize a "path" towards the learning goal.

3.3. Capabilities Associated with a Competence-Base

For each competence-base, we need to describe depths of understanding. In his revised version of Blooms taxonomy, Krathwohl [4] works with six discrete categories as attributes in a learning-taxonomy: remember, understand, analyze, apply, evaluate, and create. We adopt this approach and depict learning as adding value to a competence base.

Figure 3. Learning Seen as Adding Value to a Competence-Base.

Please note also the implicit statement here: learning in category four (apply) is only possible when building upon category three competence (analyze)!

Krathwohl made some effort to create a definite and clear description of these categories. They are key to a concise application of the model so that different parties share the same understanding. For each category, he listed verbs that should be employed when describing one category – here named competence-level. In the category "apply" for competence-base "static equilibrium of forces", the statement describing the competence of a learner could be:

"The student can build the system of ordinary equations for the bearing forces that describe the equilibrium condition of the body."

A selection of other verbs suggested by Krathwohl describing this category are

- apply
- choose
- construct
- develop
- ….

Along this line of thought learning and learning strategies can be tutored by an AI driven tutoring system: the knowledge-model that we create provides distinct decision opportunities, e.g. in terms of adding value to a competence-level or addressing a neighboring competence-base.

Figure 4. A Decision Opportunity: Should Learning Add Value to a New Base or Increase the Value of a Previously Built Competence-Level?

We now have the three ingrediencies in place that are required to represent our knowledge-concept in a computer algorithm:

1. we have proposed "competence-based learning" as a concept for the teaching methodology that our knowledge-model intends to support by conveying structured information of knowledge,
2. we have the competence-bases and their inferences defined that represent the qualitative items in the learning landscape, and
3. we have the competence-level adopted from Krathwohl [4] so that we can consistently describe a depth of understanding for each competence-base.

4. Representation of a Concept of Knowledge

Our concept of knowledge employs associated quantifiable objects. Learning and knowledge are linked by interpreting learning as an increase in competence-level. Our concept should be able to represent subjects of very different complexity – not only in engineering mechanics – to be useful. But at the same time, it should build upon a very limited number of classes of objects in order to be manageable and user-friendly.

As for all models, it serves a specific use in its abstraction of the world. It simplifies real physical, social or economic systems and there can be different appropriate models to serve one purpose. The challenge is usually to make the model as simple as possible and as complex as necessary.

4.1. Usability Criteria

Criteria for the usability of our knowledge-model have been clearly phrased in Paquet [23] as

- simplicity: it should be intuitive and easy to operate by untrained users,
- generality: it should not be tailored for one individual thematic area only (as mechanical engineering),
- completeness: it should not prohibit the representation of essential elements of knowledge due to missing types of competence-bases or relations,
- ease of interpretation: it should have a concise and plain structure so that the meaningful context of the model is self-explanatory without ambiguity,
- standardization and communicability: it should support the intuitive interpretation and the exchange of contents among users, and
- computability: it should allow an AI or servers for educational resources to be able to readily access the information contained.

4.2. Nodes

In adopting elements of a particular approach, we aim for simplicity rather than philosophical rigor. First, we follow Romiszowski [26] in his approach from instructional design to assign knowledge-items to be either factual or conceptual (abstract).

E.g. in mechanical engineering, an essential knowledge-item is the "freebody-diagram" – a pictorial representation of the forces and moments acting on a body. It is usually attributed to Joseph-Louis Lagrange who addressed it in part I (Statics), Section IV, §1 of his Mécanique Analytique. In engineering, the "freebody-diagram" would be considered conceptual knowledge, the information about author, publication etc. would be considered factual. For our knowledge-model, we will not account for factual knowledge. It only accounts for conceptual knowledge. This implies that it holds e.g. instances of principles, axioms, and rules. Thus, it is a container for the understanding of a knowledge-domain but cannot be employed to solve actual engineering problems that require factual knowledge of the systems as mass, dimensions, gravitational acceleration etc. Following Romiszowski [26] and Paquet [23], we further subdivide conceptual knowledge into

- concepts: they represent key objects in the knowledge domain, here coordinates, equilibrium conditions, forces, strains, stresses, eigenfrequencies etc.
- procedures: they represent operations on objects, here the solution process for problems in statics, the computation of principal stresses in a continuum etc. and
- principles: they represent undisputable statements, here axioms, principles, nomenclature, naming or color-conventions including ISO-norms etc.

These three are accounted for and will represent elementary types of nodes in our model. Also note that sometimes (e.g. in engineering literature, the "freebody-diagram",

in German: "Lagrangesches Befreiungsprinzip") is either attributed as an axiom or a principle. We will not differentiate between these two in our model and use "principle" as a representation for both. Later we will graphically represent the three competence-bases as a circle with the circle-color indicating its type, i.e.

● : concept,
● : procedure and
● : principle.

4.3. Edges

Relations between our competence-bases – the edges – will be represented by a directed arc, e.g.

P   : relation of precedence-type.
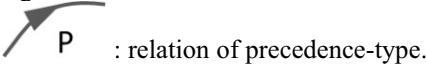
Following Paquet [23], the relational objects between nodes have each one out of six possible types, namely

- I-Link / Instantiation: the Principle of Virtual Displacement is an instance of the Principle of Virtual Work,
- C-Link / Composition: the concept of "force-equilibrium" has the "force"-concept as a component,
- S-Link / Specialization: a 2D-problem is a specialization of a generic static equilibrium problem,
- P-Link / Precedence: a "freebody-diagram" precedes the "static force equilibrium" in a solution process,
- I/P-Link / Input/Product: a quantity and vector are input to the procedure of "scalar multiplication", the product is a vector, and
- R-Link / Regulation: the axiom on "equilibrium conditions for a rigid body subject to two forces" regulates the concept "static force equilibrium".
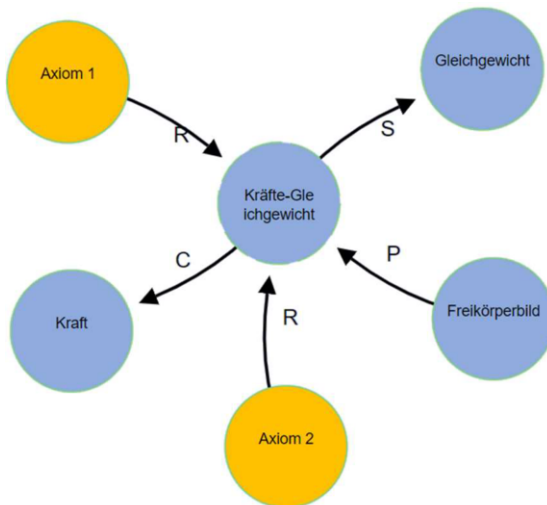

Figure 5. A Small Section of the Model.

In AI-literature, this approach to represent knowledge as linked graphic elements has been introduced under the name "semantic networks" (Hulpus [18]). Built upon the nodes of this network – or the competence-bases – is a taxonomy of detailed descriptions of skills in the form of prototypes for discrete competence-levels – from "remember" to "create". In the same way, the competence of a student can be diagnosed and quantified from appropriate tests providing a measure for the mastery of a certain subject.

5. The Web-Interface for Knowledge-Domain Experts and Tutoring Agents

This section describes the requirements and consequently implemented features of an application that we propose as an exchange-platform between teachers and AI. Due to the interactive nature of its key-function – to provide an exchange of structured information between multiple actors – this application is web-based.

It serves two purposes:

- provide an interface for knowledge-domain experts to facilitate the collaborative creation of a consistent knowledge-model,
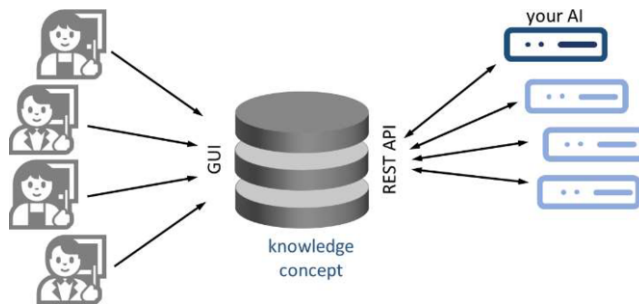- provide an interface for the interoperability with other web-based services on the Internet.

Figure 6. The database as exchange-medium between teachers and Web-based clients.

5.1. User-Interface

This section will describe the features of a user-interface to our proposed knowledge model for teachers. Our main success-scenario is that a teacher

- identifies, generates and describes a set of clearly demarcated competence-bases including their relations,
- describes each competence level in detail – including references to e.g. text-"books" (wikis) or literature and
- links the additions made to already established and agreed competence-bases.

To facilitate this scenario, the software development has been based on several use-cases. We started from the point of view of engineering mechanics – the knowledge domain that serves as the crystallization point for this application – which employs graphic content and mathematical formulas in abundance. It also turned out that users spend most of their effort on declaring and editing prototypes of competence-level-definitions. A key requisite of the software is therefore to support the unambiguous declaration of these levels, i.e. to create a distinct manifestation of a competence on explicit prototype levels. The goals that were identified are

- facilitate a simple and transparent collaboration of individuals,
- make for an intuitive, easy editing experience that allows for text-formatting, images, formulas (LaTex), and links,
- provide featured support for teachers in describing competence-levels - e.g. by proposing appropriate verbs or examples per level-prototype,
- implement a roles & rights concept to protect and allow ownership of the created contents
- allow individuals and organizations to control the extent of model-transparency and collaboration,
- have a staged approval process installed that allows content classed from "sandbox"- to "master"-level,
- aim for a generic knowledge-model that can be applied to many domains – not only engineering mechanics,
- provide a visual graph of the conceptual model that can be controlled w.r.t depth of representation (number of relations from reference-base) / contributors and
- allow for a standardized exchange of the knowledge-model (upload / download of XML-files)

```
1 package de.knowledge.concept.domain.competence
2
3 import de.knowledge.concept.domain.Application
4 import de.knowledge.concept.domain.Level
5 import de.knowledge.concept.domain.reference.Information
6 import de.knowledge.concept.domain.reference.Section
7 import de.knowledge.concept.domain.security.User
8 import de.knowledge.concept.enums.CompetenceTag
9
10 abstract class Competence {
11
12     String name
13     String description
14     String shortDescription
15
16     User user
17     Section section
18     CompetenceTag competenceTag
19
20     static hasMany = [
21         applications: Application,
22         levels: Level,
23         informations: Information,
24
25         sources: CompetenceRelation,
26         targets: CompetenceRelation
27     ]
28
29     static mappedBy = [
30         sources: 'target',
31         targets: 'source'
32     ]
33
       .
       .
65 }
```

Figure 7. Abstract Grails Domain for Competence-Bases.

A prototype of the application was first done in Protégé [24] to define the basic classes, their dependencies and understand procedural interactions. The Web-application was then implemented in Grails.

Core of the application is the definition of the abstract competence-base domain (see Fig. 7). The individual types of competence-bases extend this abstract domain, e.g. "Principle" extends "Competence". The competence-relations are declared by a domain employing different types, e.g. type "specializes".

5.2. Features

Creating a knowledge model is a very time and resource consuming process. Accordingly, it is of crucial importance that models, once they have been created, can be saved, exchanged and further developed. Therefore, as a basic feature, our editor allows to import (upload) a competence model in XML-form or export (download) the competence-model presently stored in the database.

Furthermore, as formulas are essential ingredients to describe some knowledge-domains, we have also created the option of embedding formulas directly in the textual descriptions using LaTex with a %{latex(.)}-tag, e.g. "%{latex("\vec{F}")}" as depicted in the edit-page of a Principle:

Name * Axiom 1

Description Formal bedeutet dies zweierlei (vgl. Bilder 1 und 2): Die Vektorsumme aus %{latex("\vec{F}_1")}, %{latex("\vec{F}_2")} und der Abstand a müssen

Thus, in a list of concepts, the "force-equilibrium" would appear like this:

Name	Description	Short Description
Kräfte-Gleichgewicht	Aufstellen der Gleichgewichtsbedingungen für alle Kräfte und Momente: $$\sum \vec{F}_i = \vec{0}$$ $$\sum \vec{M}_i = \vec{0}$$	Kräfte- und Momementengleichgewicht

6. An Excerpt from the Ontology

Complex conceptual models could explain the whole world – but would make them very difficult to implement and populate. Therefore, we have explicitly targeted an as basic as practical model which provides only few node- and edge-types. As a result, it is rather simple to elaborate the knowledge-model for a single knowledge-domain. Though basic, the software GUI allows for an intuitive approach. And thus, models usually grow rapidly.

6.1. Building your Knowledge-Domain

Assume you are an editor of the ontology and start from scratch with an empty database. You would choose a central competence-base as e.g. "force-equilibrium" and add it as competence-base #1. What other competences depend on "force-equilibrium"; what competences are needed to understand it?

Obviously, "force-equilibrium" requires "force" and we add "force" as a concept to the database. Since "force" is part of "force-equilibrium" we employ the "composition" or C-Link between the two. In our user-interface, you would see two blue circles connected with an arrow, all three elements being linked to objects in the database. Next comes the concept of "freebody-diagram" – which visualizes the impact of bearings or gravitational forces on a body by drawing respective forces and moments. Both – forces and moments – are represented as red arrows in the diagram in accordance with specific rules. Without the freebody-diagram, forces would not be "visible" and can not be employed in a force-equilibrium – so the freebody-diagram is a pre-requisite to the force-equilibrium. We connect the two concepts with a precedence or P-Link from freebody-diagram to force-equilibrium. The red arrows in the (factual) freebody-diagram are a special representation of a concept that we have defined already: force. We define "graphical representation" (of force) as a component of the freebody-diagram – again using the C-Link. And the graphical representation of force is a specialization of the concept of force.

So far we are only using concepts as competence-bases. A principle becomes a necessary ingredient when we deploy knowledge-items that regulate the force-equilibrium. That principle shall be "axiom 1" in statics and we link it to the force-equilibrium using a R-Link.
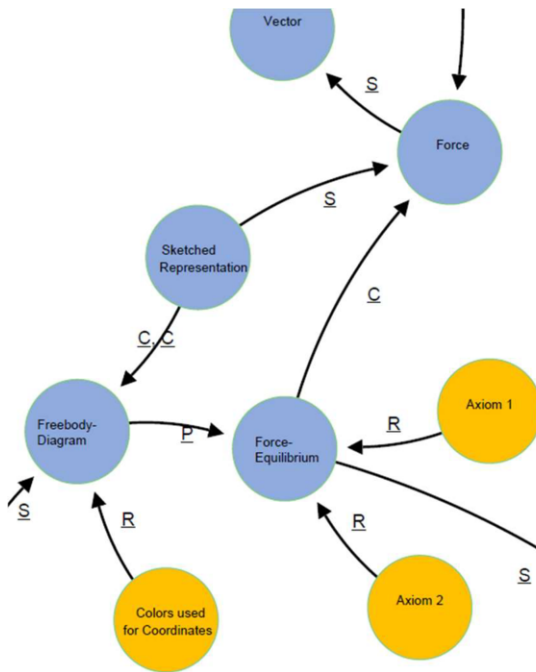

Figure 8. A Section from Our Knowledge-Model.

And another aspect enters our considerations as we construct "force". For engineers, a force has a magnitude, direction and orientation (in space). In mathematics, it would have its equivalent in a "vector", so force specializes vector. Let us assume that vector is a concept already defined in our database by a colleague from mathematics. We link it by applying an S-Link from "force" to "vector". We would learn that "vector" is already defined as the scalar multiplication of a "quantity" and a "unit vector". The new green circle "scalar multiplication with a unit-vector" in our model is thus a process taking "quantity" and "unit vector" as input and which delivers a generic "vector" as its product. In turn, a quantity would be composed of concepts "number" and "unit of measurement"; and so forth.

What appears to be a straightforward approach to create a model of knowledge in mechanical engineering has its hidden difficulties. In the example from Fig. 8 we have not included a principle that explains what "force" is. You could turn to the concept of stresses to explain force – but then it is advisable not to explain stress using force to avoid circular references.

And the precedence-links also creates an order, a process to be followed as described using "freebody-diagram" and "force-equilibrium" above. Is that a process-competence-base (a solution schema for mechanical engineering) or does it come "automatically" from an agent following the precedence-links in the ontology?

Note that guidance for the creation of a knowledge-model could also come from known student-misconceptions (Streveler [29]) which should be reflected in the structure of the model.

Finally, the main task following the creation of our semantic network, is to describe prototype-levels (Krathwohl's categories) of competence per base. Following the concepts of Competence-Based Learning, the levels must be described so they can be identified unambiguously. The software supports this process e.g. by providing a choice of appropriate verbs from Krathwohl per competence-level and the possibility to add pictures, formulas, links etc. The description must identify learning objectives as clear and concise as possible and allow for the observability of the respective actions of a student on the respective level.

6.2. Consolidation of Models

Along this line, the ontologies from different contributors grow into a structure of significant unconsolidated complexity. This raises the new challenge of elaborating a consensus-model: Just as two lectures on the same topic are never identical, the semantic models of two knowledge-domain experts will never be exactly identical either. Nevertheless, a common structure for the knowledge should be present in both approaches that can be formalized in the model. A process – and a supporting technology – is needed to facilitate this consolidation of the model. In our approach, each competence-base is therefore created with a "sandbox"-status. The database may hold incongruent nodes and these can be merged and edited by selectively displaying the ontologies of different users. Then, the role of a "competence-master" is to moderate a consolidation of these sub-models.

The representation of a consolidated ontology composed from different contributors could look like this:
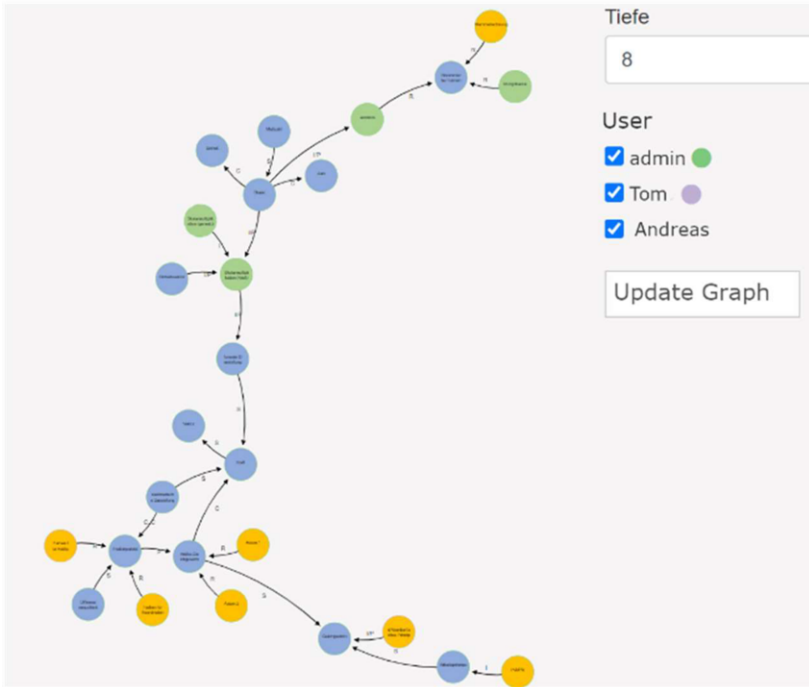
Figure 9. More Bases Shown from the Model.

7. Next Steps

Our knowledge model provides a consistent and structured model to collect and represent competence across knowledge domains. It provides a web-based user-interface suitable for phrasing competencies which allows for collaboration between individuals and organizations. Our implementation of the ontology as a semantic network in a web-application can provide a generic interface for AIs aiming to operationalize tutoring as proposed by learning experience platforms.

These models are uniform and interchangeable, i.e. as part of a community project. Basic ontologies could be created that future users can continue to develop. Catalogues are now able to add generally agreed annotations to their resources so that AI applications can parse effectively through their archives, like advanced search engines using the semantic web. Knowledge-servers as ours could also be key to future learning approaches that help tutoring agents in filtering out those bits of information that really help students to solve a difficult task (Herrman-Werner [16]), competently feed-back the findings of tests, create an individual task list to master a given competency, or even to check a curriculum for consistency.

Thus, the next step will be to employ this ontology in tutored learning environments and collect data – based on the objective progress in learning or on the subjective satisfaction of a learner (Edström [11]). Obviously, Learning Analytics using tests that address a specific competence level or simply ask for the current state of mind of a student will have an integral part in this.

This "generation zero"-tutor has to be based on explicit rules because initially, no data are available to train an AI-based agent. But as data is being collected from real learning processes, an AI could enter the stage – with a basic recommendation algorithm to begin with.

But how would they address learning? Will they aim to facilitate effective learning? Will they aim to motivate students to learn? What opportunities emerge to provide qualified feedback to students (Hartung [12])? How can the now transparent structure of the network be used to create incentives for computerized learning environments (Bartholomé [5])? Questions that can be addressed differently by different tutoring systems but based on an identical knowledge-model.

With our ontology in place, tutoring-AIs would also decide upon the fitting pedagogical approach (e.g. Schunk [27]) to teach a specific competence depending on the competence-type: a concept requires a different approach than a principle or a process! And a concept that specializes another concept may be "taught" first before offering the more generic approach – or the other way round. Also, the type of competence regulates the type of test strategy to follow, and the inference-type between bases regulates, how to address the subject. Finally, established learning materials as "Tutorials" (Riegler [25], Steinberg [28]) are an excellent basis to build a tutoring approach upon.

Our ontology provides all the pre-requisites to facilitate these developments.

References

[1] Adams WK, Wieman CE. Development and Validation of Instruments to Measure Learning of Expert-Like Thinking. International Journal of Science Education, (2011) 33(9):1289–1312.

[2] Aleven V, Mclaren B, Roll I, Koedinger K. Toward meta-cognitive tutoring: A model of help seeking with a Cognitive Tutor. International Journal of Artificial Intelligence in Education, 16 (2), 101-128 (2006).

[3] American Educational Research Association, American Psychological Association, and National Council on Measurement in Education (2014). Standards for educational and psychological testing. Washington D.C.

[4] Anderson LW, Krathwohl DR. A taxonomy for learning, teaching, and assessing. A Revision of Bloom's Taxonomy of Educational Objectives, Pearson new international ed. Harlow: Pearson Education (2014).

[5] Bartholomé T, Stahl E, Pieschl S, Bromme R. What matters in help-seeking? A study of help effectiveness and learner-related factors. Computers in Human Behavior, (2006) 22(1), 113-129.

[6] Bloom BS, Engelhart MD, Furst EJ, Hill WH, Krathwohl DR. Taxonomy of educational objectives: The classification of educational goals. Handbook I: Cognitive domain. New York: David McKay Company, (1956).

[7] Brose A, Kautz C. Identifying and addressing student difficulties in engineering statics. (2011) In Proceedings of the 2011 ASEE Annual Conference and Exposition, Vancouver.

[8] Chi MTH, Slotta JD. The Ontological Coherence of Intuitive Physics. Cognition and Instruction, 10(2/3):249–260. (1993) Taylor & Francis.

[9] Ding L. Theoretical perspectives of quantitative physics education research. Physical Review Physics Education Research, (2019) 15(2):020101.

[10] Direnga J. Assessing the Effectiveness of Research-Based Active Learning Materials for Introductory Engineering Mechanics, Dissertation; Technische Universität Hamburg-Harburg, Institut für Abteilung für Fachdidaktik der Ingenieurwissenschaften Z-2, 2021.

[11] Edström K. Student feedback in engineering: a disciplinespecific overview and background. In Enhancing Learning and Teaching Through Student Feedback in Engineering, p 1–23 Elsevier (2012).

[12] Graesser AC, Conley MW, Olney AM. Intelligent tutoring systems. In S. Graham, & K. Harris (Eds.), APA Educational Psychology Handbook: Vol. 3. Applications to Learning and Teaching (pp. 451-473). Washington, DC: American Psychological Association (2012).

[13] Halloun IA, Hestenes D. Common sense concepts about motion. American journal of physics, 53(11):1056–1065 (1985).

[14] Hambleton RK, Swaminathan H, Rogers HJ. Fundamentals of item response theory. Newbury Park, CA: Sage Publications (1991).

[15] Hartung AB. Studie zum Einsatz von Mentoring-Programmen als Instrument struktureller Förderung für Studierende an deutschen Universitäten. Arbeitspapier, 243. Düsseldorf: Hans-Böckler-Stiftung (2012).

[16] Herrmann-Werner A, Loda T, Junne F, Zipfel S, Madany Mamlouk A. "Hello, my name is Melinda" – students' views on a digital assistant for navigation in digital learning environments; a qualitative interview study, Frontiers in Education, (2021) Vol. 5, pp. 291-297)

[17] [HOOU] Hamburg Open Online University, https://www.hoou.de/.

[18] Hulpus I, Prangnawarat N. "Path-Based Semantic Relatedness on Linked Data and Its Use to Word and Entity Disambiguation". The Semantic Web – ISWC 2015, Springer International Publishing. p. 444 (2015)

[19] IEEE Approved Draft Standard for Extensible Markup Language (XML) Schema Definition Language Binding for Learning Object Metadata," in IEEE P1484.12.3/D2, November 2019 , vol., no., pp.1-0, 5 March 2020.

[20] IEEE: Learning Technology Standards Committee (LTSC). Standard for Learning Technology - Data Model for Reusable Competency Definitions. IEEE, 2008.

[21] Madany Mamlouk A, Geick C, Lämmermann K. From Zero to Hero – New Methods for Motivating Students. In Jansen-Schulz B, Tantau T (Edt.). Excellent Teaching - Principles, Structures and Requirements, Blickpunkt Hochschuldidaktik (2018) 134, 99-111.

[22] MINTfit. Online tests for math, physics, chemistry and computer science, https://www.mintfit.hamburg/en/

[23] Paquette, Gilbert: Visual Knowledge Modeling for Semantic Web Technologies: Models and Ontologies, InformatIon science reference, Hershley, 2010

[24] PROTÉGÉ: Ontologie-Modellierung, https://protege.stanford.edu/

[25] Riegler P, Simon A, Prochaska M., Kautz C, Bierwirth R, Hagendorf S, Kortemeyer G. Using Tutorials in Introductory Physics on circuits in a German university course: observations and experiences. Physics Education (2016) 51(6):065014.

[26] Romiszowski AJ. Designing Instructional Systems. New York: Kogan Page and Nichols Publishing (1981).

[27] Schunk DH. Learning theories: An educational perspective. Macmillan Publishing Co, Inc. (1991).

[28] Steinberg RN, Wittmann MC, Redish EF. Mathematical tutorials in introductory physics. AIP Conference Proceedings, (1997) volume 399, pages 1075–1092, College Park, Maryland (USA).

[29] Streveler RA, Brown S, Herman GL, Montfort D. Conceptual Change and Misconceptions in Engineering Education. In Johri A and Olds BM, eds, Cambridge Handbook of Engineering Education Research. Cambridge University Press, New York (2014).

[30] Wood, D. (2001). Scaffolding, contingent tutoring, and computer-supported learning. International Journal of Artificial Intelligence in Education, 12.