

Ontology Engineering & Semantic Web Project Report

Panagiotis Chatzichristodoulou and Kirill Tumanov

March 27, 2014

Abstract

This work serves as a reference and a summary document for the designed Student Lifecycle Management (SLM) system ontology, based on the SAP AG framework. It gives an overall structure of the ontology and describes the main functions of student lifecycle it reflects. For better comprehension this description is coupled with following the built-it exemplar individuals. The paper ends with discussion on ontology limitations and possible future work.

1 Introduction

Ontology engineering is a field that studies the methods and methodologies of building ontologies. From an Artificial Intelligence point of view, an ontology is defined as: “explicit specification of conceptualization” [7], meaning that ontologies are formal representations of sets of concepts and the relations between them within a domain. It must be noted that even though ontologies are build to serve as formal representations of a certain problem, ontologies should be build as problem-independent as possible. Consequently, an ontology about a car of a certain brand must also be able to represent cars of other brands without any modifications. The domain of the ontology described in this paper is the domain of SLM, meaning that the ontology build creates a framework that describes the SLM system. Despite the fact that it was based on SAP-SLM [1], the ontology can represent a much wider range of student lifecycle management systems.

2 Implementation

The ontology of SLM system was designed according to the available SAP AG framework, shown in Fig. 1. The implementation itself was done in Protege 4.3.0 software in the Web Ontology Language (OWL) format. HermiT 1.3.8 was used as a means of reasoning.

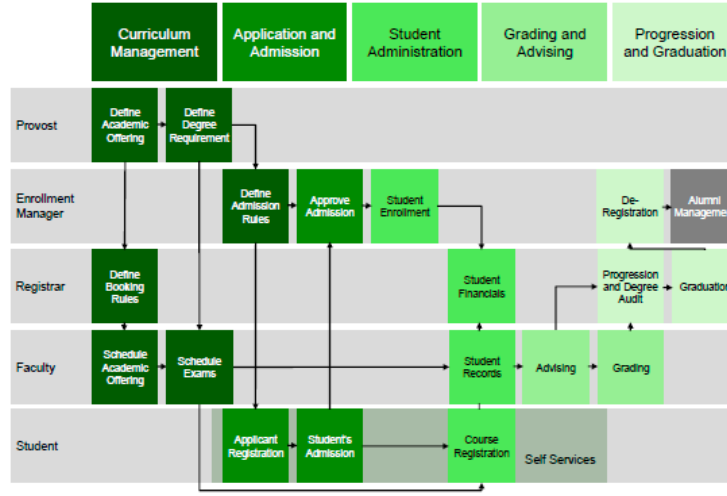


Figure 1: SAP AG framework of SLM [1]

Although there is no clear specification of what procedures the system is able to reflect, it is relatively easy to understand what are the most essential ones. Some of the blocks shown here in the ontology were logically merged or united by their roles and underlying processes. E.g. “Graduation” was merged with “De-Registration” and has a lot in common with “Define Degree Requirement”.

A more specific review of the ontology is done in the following subsections. They are organized on the built-in individuals, as a whole student lifecycle process.

2.1 Application and Graduation

Everything started when a British school graduate Bob Smith heard from his friend Joey Miller about the RWTH University, who graduated on the same year. Joey told him that some years ago he was a student at the Mathematics Department of this university. He followed a Study Programme in Mathematics very similar to the one in existence. Upon completion of the studies, he was awarded a BSc Degree in Mathematics, deregistered by Sam Jones - an Enrollment Manager and included into the list of RWTH Alumni. Now Joey was going to attend an Alumni Meeting organized by Terry Mitchel - an Alumni Manager.

Joey inspired Bob to take the same study programme as his, since Bob was quite into Math. So, Bob filed an Admission Request, which includes all needed documents and side procedures, this request was received and fulfilled (processed) by the Enrollment Manager. After the decision was made, the Admission Request resulted in Admission Response, which was sent to Bob. Bob received the Response and found himself accepted to the RWTH. He later was enrolled as a student by the Enrollment Manager.

2.2 Study Programme

Bob being a Math student selected to pursue a Study Programme in Mathematics, which culminates with an award of BSc Degree to the graduates. The Study Programme defined by Lilian Marshall - a Provost, apart from the rest includes Courses in Linear Algebra and Statistical Theory. The Programme and all its Courses are taught in English. In addition, the Study Programme has a number of Specializations, among which are a Major in Mathematics and a Minor in Statistics.

Graduation Requirements for each Study Programme are defined by the Provost. These requirements mainly include the number of Credits a Student must obtain for the Courses he/she completes successfully.

A certain amount of Credits is given for each Course, while it also is awarded to a Student upon a successful completion. A Course in Statistical Theory has a Linear Algebra as a prerequisite. Each Course is taught by a Faculty. Course is comprised of Lectures.

Each Lecture being an Event has its location, start and end time, which are scheduled by Kyle Nillson - a Scheduling Officer. A Person registered for the Course the Lecture is of may attend it. Each Lecture may have a different teaching method: a lecture, a practical or a seminar.

A Course may have one of the following statuses: booking, ended, fully booked or ongoing. When Course is in a booking state, a Student is able to, first, book a place in it and later to be registered. An end of a booking term is defined by the Course's booking deadline. Finally, after taking a Course a student may fail it or complete successfully.

Successful completion or failure of a Course is based on the Student's Grade obtained at the corresponding Exam. Exam is an event, like a Lecture, and thus has similar features. In addition, an Exam Paper is issued for each Exam, meaning a set of questions a Student is to answer. A Faculty member conducts an Exam, grades the Exam Paper and reports a Grade to the Student.

Bob's Linear Algebra course is taught by Jeffry Hunt - a Full Professor of Mathematics, and a Statistical Theory course by Samantha Kole - a Senior Lecturer. He is registered for both of them, and attends the first and second lectures of each one respectfully.

2.3 Student Administration

A Student being enrolled at a University is obliged to pay Tuition. This part of student finances a University is concerned about. So since Bob got enrolled he was a subject of tuition payment check.

The check is a part of the Tuition Management service of the University. In RWTH Student Administration office provides this Service, with Frank de Boer - a Registrar being responsible for it. However, as Bob just got accepted the first check was performed by the Enrollment Manager, who filed the tuition fees request and sent it to Jessica Vervier - a University Accountant. She checked the RWTH public account to see that the tuition fees were paid, thus fulfilling

a request, which resulted in a positive response sent back to the Enrollment Manager.

However not only finances but all the personal information and grades of a Student are in the competence of the Student Administration office. This data is stored and retrieved from the RWTH grade and personal info databases, accessible through a University Portal in the Web. RWTH Students and personnel have different privileges using the Portal.

2.4 Advising

Once Bob became a Student he started to take care of his upcoming Internship. As he was new to the procedures he decided to ask an advise of the Mathematics Department Study Advisor - Gonny Millews. To do so Bob filed a request of additional info on the internship possibilities, which was a general e-mail on practice. Gonny received his e-mail, thought of an answer and replied, providing him a general information and useful links. Therefore, she fulfilled a request and produced a response to it.

Once Bob received this information from the Study Advisor, he contacted a company named Diamond LLC. There he was said that they had a placement - an Internship in Data Analysis. After thinking of his career prospects Bob decided to accept the offer and fill the position. Since then, in addition to being a Mathematics Student Bob also plays a Role of a Data Analyst.

2.5 Visualization

This section provides and analyses Cloud View visualizations of the ontology. Such visualizations provide insight about the relations between its T-Box and A-Box. Firstly, in Figure 2 classes that are used the most in the ontology are displayed. That is the ones that have the most relations and provide the basis in which the ontology is build. As expected, the classes of *Course*, *StudyProgramme*, *Student* and *Role* play a major role in the ontology and serve as its backbone. On the other hand, classes like *Database* and *Document* are used much less even though they are essential to the structure of this ontology.

Secondly, information about the amount of functionality each individual within the ontology has, is displayed in Figure 3. Again, instances of type *Course*, *Role*, *StudyProgramme* and *Student* are the most used ones. Furthermore instances like *UniversityAccountant* have much less connections despite being essential to the ontology. That is expected since some individuals provide the core of the ontology and are used more than others. That displays a good correlation between the Terminology of the ontology and its Application.

2.6 Conclusion

Building an ontology that successfully represents a problem domain is not a trivial task. It requires a correct abstraction of the notions included, and iterative restructuring of the ontology so that it has an correct structure. In this work the



Figure 2: Cloud view of the ontology visualizing class usage

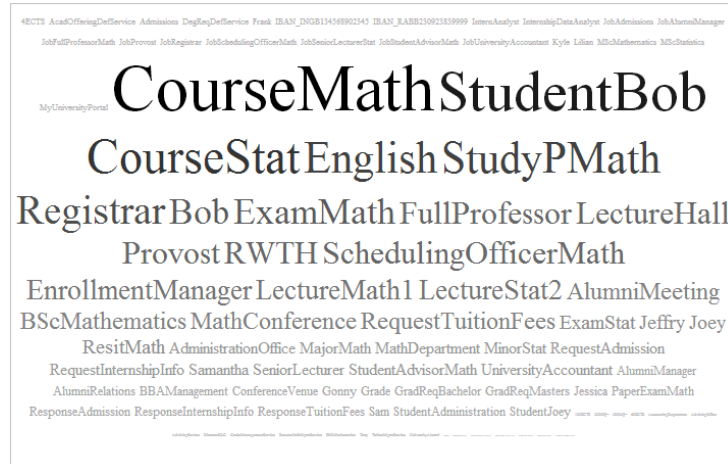


Figure 3: Cloud view of the ontology visualizing individual usage

designed ontology's capabilities were demonstrated from the application point of view rather than the structural one. This approach was chosen because it makes the tasks of reporting and describing the usage of the ontology easier and more complete; easier because it removes from the reader the cognition load of finding usage for the ontology, and more complete because it displays an A-Box and, as a result, completes a basic Knowledge Base of the ontology.

References

- [1] Solution in Detail: Higher Education and Research. Student Life Management. SAP AG, <http://www.sap.com/bin/sapcom/en.us/downloadasset.2013-12-dec-11-11.higher-education-and-research-student-lifecycle-management-pdf.html>(2013)
- [2] Protege ontology creation tool Website. <http://protege.stanford.edu/>
- [3] Protege ontology creation tool Wiki. http://protegewiki.stanford.edu/wiki/Main_Page
- [4] Protege tool cloud views plugin. http://protegewiki.stanford.edu/wiki/Cloud_Views
- [5] Protege onto graph views plugin. <http://protegewiki.stanford.edu/wiki/OntoGraf>
- [6] Ontoclean Method. http://www.researchgate.net/publication/27293101_Evaluating_ontological_decisions_with_OntoClean/file/9fcfd50-f9621161392.pdf
- [7] Gruber T.: [http://www-ksl.stanford.edu/kst/ what-is-an-ontology.html](http://www-ksl.stanford.edu/kst/what-is-an-ontology.html).