



Co-funded by the
Erasmus+ Programme
of the European Union

An abstract graphic consisting of several vertical, rounded rectangular bars in red, yellow, green, and pink, arranged in a staggered pattern. Below these bars is a blue silhouette of a hand with fingers spread, reaching upwards.

Description of the public APIs for the integration of 3rd party tools in BLOOM

Description of the public APIs for the integration of 3rd party tools in BLOOM

This report is the deliverable of D2.14 of the OpenU project

By J. Posel, with the support of H. Steller and A. Kovalevski, Freie Universität Berlin

Co-funded by the
Erasmus+ Programme
of the European Union



Disclaimer: Funded by the European Commission in the framework of the Erasmus+ co-funded project OpenU (606692-EPP-1-2018-2-FR-EPPKAC-PI-POLICY).

The European Commission support for the production of this publication does not constitute an endorsement of the contents which reflects the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

License used:

This work is licensed under a Creative Commons Attribution 4.0 International License:

(<https://creativecommons.org/licenses/by/4.0/>) Anyone is free to **share** (copy and redistribute the material in any medium or format), and **adapt** (remix, transform, and build upon the material) for any purpose.

Table of contents

1. Public APIs for the integration of 3rd party tools	3
2. Synchronizing between components of the BLOOM Hub	3
3. Future considerations within the ecosystem	7

Note: this report summarizes in chapter 2 the findings of Artmen Kovalevski regarding the “Analysis, development and integration of software components in SAKAI and MyCampus in context of the shared European digital infrastructure Bloom Hub”.



1. Public APIs for the integration of 3rd party tools

The BLOOM Hub has been envisioned as a modular platform, assembling components that can both bridge between existing systems, as well as creating synergies by connecting to or facilitating integration of other services.

2. Synchronizing between components of the BLOOM Hub

By its own nature, the BLOOM Hub makes data accessible towards its different modules. For example, course catalogues are collected, enriched and aggregated within the LPS system, then transferred towards the course catalogue as well as the LMS component of the Hub via an import in structured XML format. A sample of an import file in the structured XML format is shared in the annex and partly shown in Fig. 1.

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <courses semester="WS 22/23">
3   <course status="mvs.PUBLISHED">
4     <courseId>5071ad4f-4846-4c0e-8747-c814617920df</courseId>
5     <title>Title of main lecture</title>
6     <description><lt;p>Information about the content of them main lecture.</p></description>
7       <#xD;
8       <lt;p><em>Free text (HTML formatted)</em></p></description>
9     <readings><lt;p>Literature and other resources.</p></readings>
10       <#xD;
11       <lt;p><em>Free text (HTML formatted)</em></p></readings>
12       <#xD;
13       <lt;ul></ul></readings>
14       <lt;li>Some Book</li></ul></readings>
15       <lt;li>Some Journal article</li></ul></readings>
16       <lt;li>Another book</li></ul></readings>
17     </readings>
18     <additionalinformation><lt;p>Further information about this course offering, e.g. prerequisites for
19 participation.</p></additionalinformation>
20       <#xD;
21       <lt;p><em>Free text (HTML formatted)</em></p></additionalinformation>
22   </course>
23   <lecturers>
24     <lecturer>
25       <email>email.of.lecturer1@hei.edu</email>
26       <firstname>firstnameLecturer1</firstname>
27       <isHeadLecturer>true</isHeadLecturer>
28       <lastname>lastnameLecturer1</lastname>
29       <userNameZedat>pairwise-id of lecturer 1</userNameZedat>
30       <website />
31     </lecturer>
32     <lecturer>
33       <email>email.of.lecturer2@hei.edu</email>
34       <firstname>firstnameLecturer2</firstname>
35       <isHeadLecturer>false</isHeadLecturer>
36       <lastname>lastnameLecturer2</lastname>
37       <userNameZedat>pairwise-id of lecturer 2</userNameZedat>
```

Fig. 1: Sample view of a import file formatted in structured XML.

To further simplify the processing of the different data sources, an extension to the BLOOM Hub, the KVV Manager, has been adapted. KVV Manager is a tool, created and maintained by the MyCampus team at Freie Universität Berlin, that exports course data to Whiteboard from Evento, a local IT system. This tool directly creates sites using course data from Evento, by taking XML-generated course data from Evento and sending it directly to Sakai, fetching it from the Evento server by a dedicated link. There is also an extra option of uploading an archive with the relevant data. The same entry point can be used to allow third parties, for example by facilitating an export of courses to be shared with a broader target audience from an existing local course catalogue.

In general, SAKAI doesn't provide an API that creates Sites from scratch. While there is a tool to create individual sites by hand, it does not operate on serial basement. SAKAI sites are meant to be created on

the base of the Course Management System, which is a proprietary course data organization in Sakai. As a BLOOM Hub prototype during development, this Sakai version should provide a course overview for multiple universities across Europe. Therefore, it was very important to develop standard ways of exporting course data. Course management is a standard built-in solution of organizing course data and site creation in Sakai and suits the requirement of standardization.

Fig. 2 shows the data model used to store and handle objects within the LPS system. The relevant objects would be:

- Faculty
- Department
- Semester
- Abstract Course / Abstract Ancillary Course (contains course data, that are independent from semester)
- Course / Ancillary Course (instance of Abstract course/Ancillary during specific semester)
- Series (Set of periodical events of courses (example: lecture on Monday from 8:00 to 10:00, each week of semester)
- Appointment (individual event of course (lecture on Monday, 10th October, 8:00-10:00)

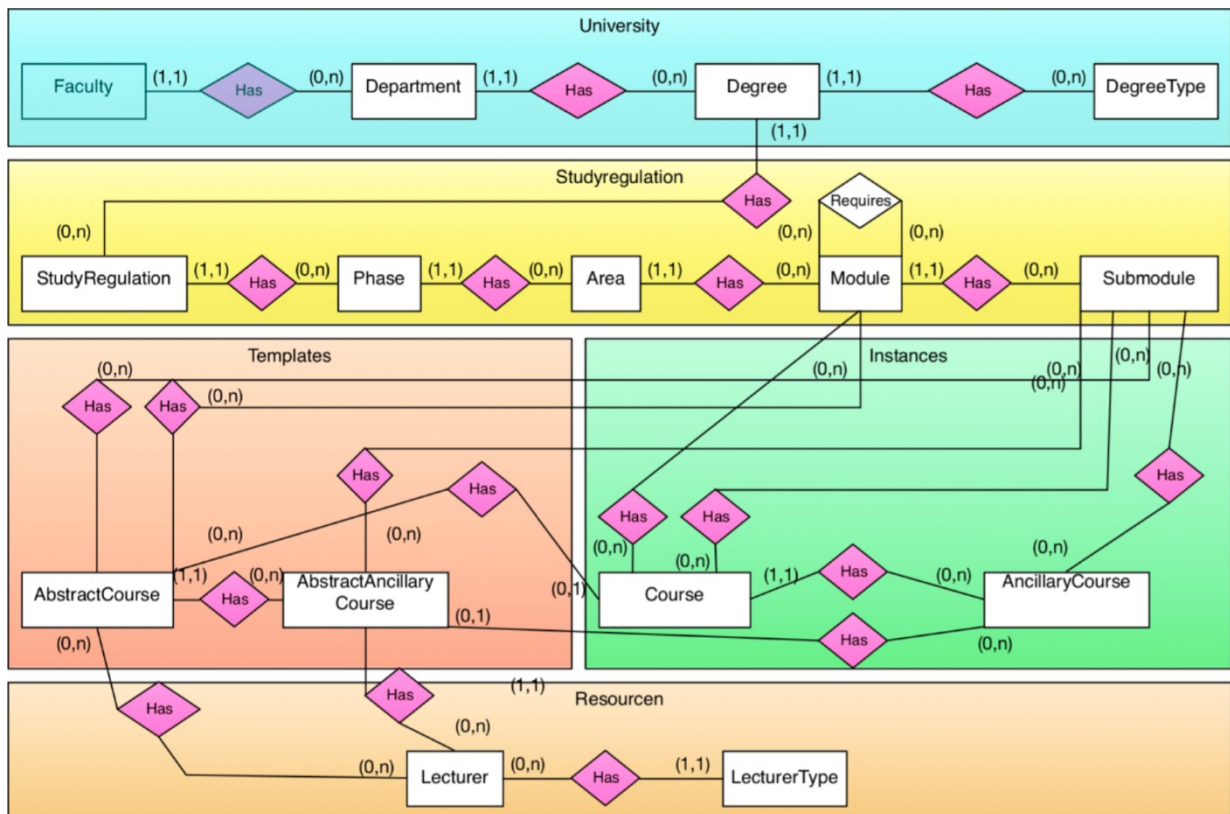


Fig. 2: ER Diagram Chen-Notation with Min-Max-Notation, Author: Stephan Sundermann

In comparison, Fig. 3 shows the data structure used by the Sakai Course Management System. There relevant objects are:

- Course Set (Set of Courses or Canonical courses united by any possible characteristic)
- Academic Session (big temporal cycle for study process, like semester)
- Canonical Course (course data, independent from academic Session)
- Course Offering (Canonical Course instance during certain semester)

- Enrollment Set (used for relations between users and course offerings (students, lecturers of certain courses))
- Section + Section Category (set of events for course offerings with categories (lecture, tutorial, etc.))
- Meeting (weekly time table data for sections)

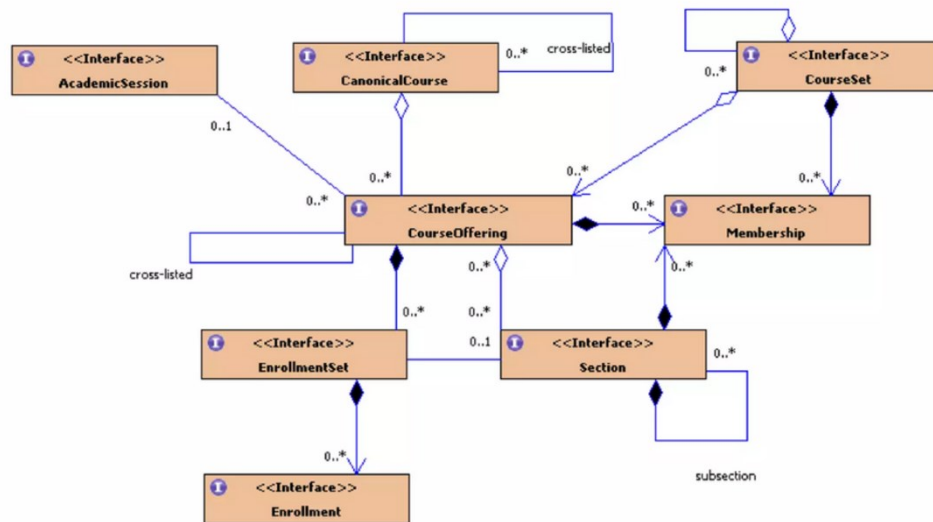


Fig. 3: Source: Sakai Wiki, Educational Services, <https://sakaiproject.atlassian.net/wiki/spaces/DOC/pages/17225318818/Educational+Services#>

Based on both sets, a conversion scheme has been developed, as shown in Fig. 4. Furthermore, possible methods of course data transfer, like upload of CSV data (Quartz jobs, DirectAPI REST functionality), an API for programmers (Sakai beans) or On Fly Fetch (Rewriting Course Management Interfaces) have been weighted and discussed (cf. Deliverable 2.7 for more in-depth details)

The implementation of the API approach provided a solution with basic functionality: most relevant data is present, like university/faculty/department relations. The integrated tool has access to basic data in order to create basic sites. However, the solution is not complete. Due to time constraints only implementation of basic functionality was accomplished. The newly created sites contain no description (although description data is present in canonical course data, as well as in course data), no participant information, no calendar appointments. In general, basic functionality is present, and the proof of concept is considered successful.

Class 1	Type 1	Field 1	Type 2	Field 2	Class 2
Meeting					Appointment
	Section	section	Series	series	
	String	location	Set	rooms	
	String	eld	Integer	a+appointmentID	
	String	title	String	name	
	String	description	--	--	
	String	notes	--	--	
	String	authority	--	--	
	Time	startTime	LocalDateTime	startTime	
	Time	finishTime	LocalDateTime	endTime	
	boolean	monday	int	getDayOfWeek()	
	boolean	tuesday	int	getDayOfWeek()	
	boolean	wednesday	int	getDayOfWeek()	
	boolean	thursday	int	getDayOfWeek()	
	boolean	friday	int	getDayOfWeek()	
	boolean	saturday	int	getDayOfWeek()	
	boolean	sunday	int	getDayOfWeek()	
Section					Series
	String	eld	Integer	s+seriesID	
	String	title	String	appointment.getTitle()	
	String	description			
	Section	parent	--	--	
	String	category			
	Set	meetings	Set<Appointment	appointments as Meetings	
	CourseOffering	courseOffering	Course/AncillaryC	course/ancillary	
	Section	parent			
	EnrollmentSet	enrollmentSet			
	Integer	maxSize	Integer	room.getSize()	
SectionCategory					
CourseOffering					Course, AncillaryCourse
	String	title	String	name	
	String	eld	int	k+number nk+number	
	String	status	String	ancillary/courseStatus	
	String	description	String	description	
	AcademicSession	academicSession	Semester	Semester	
	Date	startDate		startDate of earliest Appointment	
	Date	endDate		endDate of last Appointment	
	CanonicalCourse	canonicalCourse	AbstractCourse	ak/ank+abstract Ancillary/Course Number	
	String	canonicalCourseEid	Integer	ak+abstractCourseNumber	
	Set	courseSets	Department	department	
	Set	courseSetEids	String	d+departmentNumber	
CanonicalCourse					AbstractCourse
	String	title	String	title	
	String	eld	Integer	ak/ank+abstractAncillary/CourseNumber	
	String	authority			
	String	description	String	description	
	Set	courseSets	Department, Faculty, University		
	Set	courseSetEids			
	CrossListing	CanonicalCourses, CourseOfferings	--	--	
CourseSet					Department
	String	eld	String	d+departmentNumber	
	String	title	String	Department name	
	String	description	--	none	
	String	authority			
	CourseSet	parent	Faculty	faculty	
	String	category	--		
	Set	courseOfferings			
	Set	canonicalCourses	Set<AbstractCourse>+Set<AbstractAncillaryCourse>		
CourseSet					Faculty
	String	eld	String	f+facultyNumber	
	String	title	String	Faculty name	
	String	description	--	none	
	String	authority			
	CourseSet	parent	U	University	
	String	category	--		
	Set	courseOfferings			
	Set	canonicalCourses	Set<Department>	departments	
CourseSet					University
	String	eld	String	fuberlin	
	String	title	String	Free University of Berlin	
	String	description	--		
	String	authority			
	CourseSet	parent	--	null	
	String	category	--		
	Set	courseOfferings			
	Set	canonicalCourses	Set<Faculty>	faculties	

Fig. 4: Conversion scheme between Objects, Source: Documentation of MyCampus Team

3. Future considerations within the ecosystem

There are several considerations to be taken into consideration for the future development of the BLOOM Hub, mainly driven by evolving feature requests and requirements originating from the exchanges and workshops with members of European university alliances.

One key anchor is the topic of self sovereign identities (SSIs). In the realm of the European higher education are, the SSIs are addressed by the European self-sovereign identity framework (ESSIF), which itself is part of the European blockchain service infrastructure (EBSI).

The EBSI is a joint initiative from the European Commission and the European Blockchain Partnership (EBP) to deliver EU-wide cross-border public services using blockchain technology. Katholieke Universiteit Leuven and the Università di Bologna, both partners in the Open U project as well as the Una Europa European university alliance, are involved in the European Commissions EBSI initiative with the implementation of the exchange of verifiable educational credentials between institutions in cross-border settings.

Similarly, partners of the European Reform University Alliance (ERUA) are implementing a self sovereign identity called “ERUAID” to be used within their European university alliance, that is envisioned to be used as means of authentication and authorization within digital wallets, the latter being compatible with the EBSI.

Both approaches are envisioned to be supported and integrated within the BLOOM Hub, in order to respond to the needs not only of securely authenticating and validating students both versus partner institutions as well as legitimate third parties, but also of verifiable educational credentials shared between alliance partners and beyond, consequently enabling (former) students to access digital credentials possibly on a life-long time frame.

Furthermore, in regards to the ongoing evolution of the ELMO Format and the EMREX network we are following the development both of the European Digital Student Service Infrastructure Level 2 and its implications on, for example, Erasmus Without Papers¹, as well as emerging platforms like PIM², specifically geared towards credit recognition in national and international student mobility, in order to provide the best possible connectivity options within the BLOOM Hub and its community in this maturing environment.

¹ „edssi Erasmus Without Paper – Getting ready for the new Erasmus+ programme”, https://www.eunis.org/eunis2021/wp-content/uploads/sites/18/2021/06/10_1422-Janina-Mincer-Daszkiewicz.pdf, June 10th, 2021. Retrieved December 19th, 2022.

² „Platform for International Student Mobility”, <https://pim-plattform.de/en/>. Retrieved December 19th, 2022.



Project Number: 606692-EPP-1-2018-2-FR-EPPKA3-PI -POLICY

Project Duration: 47 months

Start date: 20-02-2019

End date: 19-12-2022

Coordinator: Université Paris 1 Panthéon-Sorbonne

License used: This work is licensed under a Creative Commons Attribution Share alike 4.0

International License: <https://creativecommons.org/licenses/by-sa/4.0/>

With this license, you are free to **share** copy and redistribute the material in any medium or format. You can also **adapt** remix, transform and build upon the material for any purpose, even commercially.

But only Under the following terms:

Attribution — You must give appropriate credit, provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.

Share Alike — If you remix, transform, or build upon the material, you must distribute your contributions under the same license as the original.

Disclaimer: The European Commission support for the production of this publication does not constitute an endorsement of the contents which reflects the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.



Co-funded by the
Erasmus+ Programme
of the European Union

Freie Universität



Berlin

