A Metadata Model for Online Laboratories

Per Pascal Grube, David Boehringer, Thomas Richter
Rechenzentrum
Universität Stuttgart
Stuttgart, Germany
{grube,boehringer,richter}@rus.uni-stuttgart.de

Claus Spiecker
Universitätsbibliothek
Universität Stuttgart
Stuttgart, Germany
Claus.Spiecker@ub.uni-stuttgart.de

Nicole Natho
Center for Multimedia in Education and Research
School of Mathematics and Natural Sciences
Technische Universität Berlin, Berlin, Germany
natho@math.tu-berlin.de

Christian Maier, Danilo Zutin
Carinthia University of Applied Sciences
Villach, Austria

Abstract—**Making online laboratories available to the wider public requires them to be retrievable and reusable. To this end, we define a metadata set providing all required information in a machine readable form. This article presents the metadata set currently under discussion by the Global Online Lab Consortium (GOLC) as well as the issues of defining widely acceptable controlled vocabularies to describe the scientific field of remote experimentation.**

Keywords- Metadata, Online Experiments, Semantic Web

I. Introduction

In 2009, the University of Stuttgart started an EC-funded initiative for the dissemination of Remote Experiments and Virtual Laboratories, the *Library of Labs*, or short "LiLa" [7] in the following. The goal of this project is to make remote experiment installations and simulations already present at participating partners available to a larger audience, and collect a corpus large enough to cover most of the needs of undergraduate engineering courses. Clearly, just *collecting* material is not enough, it must also become *accessible* to teachers and students, and for that, teachers must be enabled to locate the material they need. Despite issues of locating such experiments, a portal presenting and offering content for download must also clearly state under which terms and conditions its content is available, whom to contact to obtain the necessary rights, how to access – technically – the remote equipment etc. In short, a *metadata set* is required that annotates the contents in the LiLa portal.

This paper describes the design considerations of this metadata set, the ontology it is based on, the rationale for the design decisions we've taken as well as its current discussion in a world-wide interest group on remote experimentation, the *Global Online Lab Consortium*, or short the GOLC.

II. Related Work

The idea to use online labs to enrich student education and allow students the execution of experiments independent of lab hours and the availability of faculty staff has been established quite a while before the LiLa project has been initiated, of course. The likely most prominent example of an architecture for remote experimentation is the *iLabs* infrastructure of the MIT in Cambridge, MA [1], the VISIR online lab for electrical engineering [2], or the OmniPrex initiative of the TU Berlin [3]. Engagement of the authors in this field goes well back to 2002 with the implementation of a virtual laboratory on many-body physics [4]. All these early initiatives, however, were rather isolated in themselves and networking, if ever, only happened at university level. Finding and locating an online lab that is suitable for the needs of a specific lecture was, at that stage, mostly a matter of luck, and of using search-engines like *google* appropriately.

The first project that addresses this deficit to our knowledge is the Lab**2**Go project of the Carinthia University of Applied Science [5] which establishes a repository of online labs and also defined a metadata set to annotate its content.

The Lab**2**Go initiative, the Australian LabShare project [6], the iLabs project, LiLa and additional interest partners formed the Global Online Lab Consortium, aiming at international cooperation and reaching interoperability between platforms to allow the exchange of content. One of the earlier outcomes of the discussion between GOLC partners is a metadata set for annotating remote labs described in this paper.

This paper is structured as follows: In the first section, we address design considerations we have taken into account to address the individual installations and architectures. Following that, we present the ontology our metadata set is based on. In the next section, we present on some examples how our metadata set works by showing some of the elements and their practical application. Including the full set in this paper would unfortunately go beyond the scope of this paper, but the interested reader is encouraged to retrieve the full copy of the set [8].

III. Design Consideration

The design of the LiLa metadata set is determined by its use-cases and its application in the LiLa portal. Since LiLa aims at providing teachers help to find suitable online laboratories for teachers, the most important application of the LiLa portal is the *search functionality* for such interactive material. That is, the LiLa metadata set allows annotating the experiment by the scientific field and its target audience. Furthermore, maintaining remote experiments might involve permanent costs, for example due to the supply with chemical ingredients for a chemistry lab that must be covered by usage fees. Other laboratories might require proprietary software that must be licensed, and as such involves costs for the content provider as well. For this reason, the metadata set also needs to define license terms, usage conditions, and a contact to negotiate the costs. This person is not necessarily identical to the operator of the lab which can be contacted in case of technical problems, or in case the lab requires maintenance.

LiLa does, even more, not only store and maintain the labs itself, but also includes learning material, lessons and activities, for such labs; this is because developing a pedagogical sound course on an online lab is non-trivial, and sharing learning material might ease the deployment of labs. The contact person for such a course, the mediator, might again not be identical to the operator, or the licensing institution. An online course, unlike the raw material itself, also has a target audience it has been designed for; – the same technical laboratory might address varying audiences depending on the instructions and experiments that have to be conducted in it.

Remotely controlled hardware, also called *rigs* in the following, are resources that can only be accessed by a single user or cooperating group of users at once – quite unlike simulations where user counts are only limited by the server capacity. This implies that rigs must be *booked*, i.e. their usage must be reserved in advance. Specifically, this requires the metadata to uniquely identify rigs, not only the experiments that run on them (see also section IV); several experiments might require the same physical rig to operate on, or one experiment may be implemented on several rigs. Specifically, the – here still imprecisely defined – term *experiment* must be carefully distinguished from the hardware it runs on. This is the matter of a carefully defined ontology discussed in the next section.

Last but not least, one of the goals of the LiLa project is also to allow libraries to index LiLa content, i.e. enter it into library catalogues for research purposes.

The library community, however, has rather strict requirements on the metadata set to be used: The set must be tightly defined; every indexed entity requires at least an author and a publication date, and additional requirements that go beyond the scope of this article. The core set, which we base our metadata on, is thus the Dublin Core [9], plus suitable extensions necessary for LiLa purposes. Unfortunately, the choice of Dublin core as basis partially contradicts another goal of LiLa, namely offering all material available at the portal as downloadable SCORM modules [10]; SCORM, unlike library catalogues, use LOM [11] as its basis. To address this problem, the LiLa material present in the portal uses the LiLa set to annotate its content, but exporting material in the form of SCORM modules transcribes the metadata to the LOM set whenever applicable, and embeds both the Dublin Core as well as the LOM defined annotations in the SCORM module.

The LiLa metadata set is under consideration for international application by the Global Online Lab Consortium (GOLC) [12], and thus needs to address a variety of different architectures, and not only those designed for LiLa itself. This results in a couple of design peculiarities that are discussed along with the ontology of the set in the following section.

IV. THE ONTOLOGY STRUCTURE

The LiLa metadata profile is organized along the LiLa ontology, see Fig. 1:

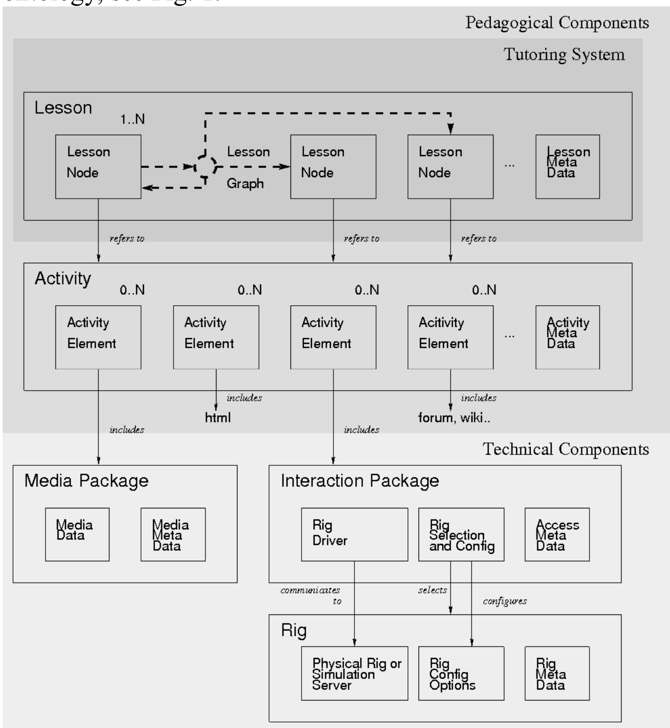

Figure 1: The LiLa Ontology

The bottom level of this ontology is formed by the so-called *Rig* which denotes the physical hardware a remote laboratory runs on; this hardware includes – besides the remotely operated physical setup – also the servers required to remotely operate this hardware. In case the rig implements a simulation, the rig consists *only* of the simulation server. In case the simulation runs on the client host only, no rig is present.

Rigs are, due to the nature of the LiLa project, remotely controllable, though the type of interface a rig implements to this end remains unspecified. Only the combination of an *interaction package,* described below and the rig enables remote usage. Besides this interface, rigs might be configurable, i.e. might provide several different experiments by reconfigurable hardware or flexible software.

Configuration and *operation* of a rig are overlapping terms and not tightly defined; the *configuration* typically happens at

the start of a session, likely invisible to the user, while *operating* a remote experiment or simulation implements the purpose of the rig. Rigs are equipped with metadata, however unlike other metadata elements, this data is not directly stored at the rig, but at a portal, for example the LiLa portal, hosting rigs. The purpose of this data is *only* to describe the rig itself, its owner, and the rights under which the rig can be accessed.
Rig metadata enables *reservation* and *booking* of rigs, i.e. while many access packages exist that implement various experiments on a rig, rigs are a scarce resource that requires careful handling as it is not available to more than a limited number of users at once.

Special care must be given to rigs that are implemented as several identical copies of the same physical setup; this type of configuration is for example found at the LabShare project. Within the ontology at hand, this collection or set of hardware is still considered a *single* rig, under an admittedly abuse of language, as it still requires booking and as its copies are accessed by the identical *interaction package*, a term more closely defined below. However, booking for such setups is relaxed in so far as several users might access the same rig at once without requiring cooperating users. Booking can, in this case, be relaxed to *queuing* where users access available setups on a first-come first-serve basis. This model is especially attractive for remotely operated simulations as servers are usually able to host more than one user at a time.

While the rig is located at the content provider, the *interaction package* is running in a web browser at the client machine. Its purpose is to provide all necessary interfaces of the rig, and make it available to the web browser of the user by implementing some kind of standard interface. This may imply that users have to install a browser plug-in, for example a java or a flash plug-in. Such *technical requirements* are annotated in the metadata accompanying the interaction package. Interaction packages also configure the rigs they control, may select one out of several copies of the same setup implementing the rig, and communicate in some proprietary way to its servers which remains, for LiLa purposes, opaque.
The purpose of the metadata here is, for first, providing some technical data as the rendering dimensions required for the user interface in a browser window, but also providing a contact for users in case of trouble with the hardware. They also provide background information on the license conditions of the interaction package itself, but not on the rig.

It is important to note that rigs and interaction packages *do not* define a didactical use-case or scenario. Both are multi-purpose objects, and the graphical representation of an access page in a browser – i.e. the GUI of the "experiment" – should be rendered *without* any accompanying text or explanation. This is not only because such background information is better placed in a higher level defining the purpose of a rig, but also because a booking system might block access to an access package, and hence may render such information invisible.

Technically, access packages within LiLa – but not necessarily for GOLC in general – are implemented as SCORM packages that bundle the metadata along with the technical interface for the browser, i.e. an HTML page containing an item or interface to the rig; the latter could be an applet or a LabView interface. These packages are deployed in a SCORM run-time environment, either that of a learning management system such as Ilias or moodle, or by the SCORM run-time environment provided by the LiLa portal. Interaction packages are only allowed to use a small subset of the full SCORM interface, as for example a user name or a user ID; this helps to keep the SCORM runtime simple; specifically, interaction packages are bare any pedagogical intent and hence must not include any metadata defining pedagogical purposes.

Media packages are similar to access packages in the sense that they provide access to traditional, less interactive media like text documents, videos or audio recordings. They consist of the media itself, represented in its natural container, and metadata describing the contents, its language and format, the rights and the owner of such media. Unlike interaction packages, media packages are implicitly created when uploading media in a known format to the LiLa portal, and metadata is only created in the upload process. Media packages may optionally be downloaded as SCORM packages, though, to ease their deployment in learning management systems.
Quite unlike access packages, media packages do not require *booking* or *reservation* and can always be accessed; however, quite like access packages, though, media packages do not define a pedagogical purpose and are only self-contained elements that might be required as a component for such a purpose. This choice of design seems curious to the alert reader at first, but one should remember that a given text document can be read under quite a different aspects and with different intentions, and these *intentions* are specified at a higher level in the LiLa ontology.

To support a pedagogical purpose, interaction packages and media packages are orchestrated into *learning activities*, which are represented within the LiLa portal as html pages delivered by its portal server. In addition to interaction and media packages, activities may contain supplementary material, as for example raw html text that provides explanations, instructions or background information; such activities can also be equipped with interactive web-contents, for example forums, wikis or mechanisms to rate the content.
Activities exist only in the form of an html page, and the content it consists of; it is, as such, an entry in the server database. Similar to interaction and media packages, activities are annotated with metadata, though this metadata describes now the pedagogical purpose, the duration, the educational level and the instructional method. Similar to the interaction and media package, rights holder and license conditions are included, but now on the activity as a whole object. The activity itself is currently not considered to be representable by

a SCORM object in general at the time of writing. Such an encoding might be feasible for some activities.

While activities are the smallest object having a pedagogical purpose, they are still represented by a static html page, though potentially one containing interactive content; that is, by its very representation form, activities focus on a single learning content to be comprehended by working through the presented material on the page, though are less suitable for complex matters that require the combination of or relation between several topics. It is the purpose of a *lesson* to represent such a process, and to guide users through the available material. Quite like activities, lessons are defined by their pedagogical purpose, but their granularity is much coarser.

Lessons may be interactive, may react on user input and may suggest a choice from multiple activities on a matter the one that is most suitable for helping the learner; they may also simply define a linear structure of activities to be worked through one by another. Nevertheless, lessons are still small units that are fit for homework assignments, and are not designed for representing full lectures, a unit not yet defined in our ontology. Such tasks are better left to established learning management systems LiLa does not attempt to replace.

Lessons are represented within the LiLa tutoring system, and thus operate on activities as their elements; lessons are not only a collection of activities, but also define the interaction between them, and the paths through the material learners may take to follow the lessons including detours, bypasses or redirections. That is, unlike activities, lessons define a *complex learning activity* taking place within the tutoring system. Lessons only exist as entries in the database of the tutoring system, and LiLa does not aim to represent them as SCORM objects. However, as the purpose of lessons and activities are quite similar, both are described by the same metadata set, namely that of an activity defining the didactical purpose, rights holder, mediator and level of a learning unit.

V. An Interaction Package Example

The LiLa Portal uses the Jena RDF[14] storage as its primary data source. One main reason for choosing RDF over a traditional database is the far greater flexibility of the stored data, and the better exchangeability of RDF. Figure 2 shows an example of an interaction package exported using the RDF/XML format.

VI. Generating Controlled Vocabularies

Most controlled vocabularies currently available are used in the library and publishing context. They are designed specifically by and for this user group and intended to be used by teachers and learners. For this reason, the LiLa and Lab2Go projects have chosen to use the categories provided by the DBpedia project [13]. The DBpedia provides the categories used by the Wikipedia as a semantically annotated skos file, thus being based entirely on input from the Wikipedia users.

Acknowledgment

This work is partially funded by the EC through the LiLa project. The authors want to thank the Computing Center of the University of Stuttgart to support the work as well, and appreciate the discussion with GOLC members to clarify the structure of the ontology. Specifically, David Boehringer and Thomas Richter want to thank David Lowe and the UTS for their hospitality and the fruitful discussion on the metadata ontology during their stay in Sydney in January 2010.

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
 xmlns:j.0="http://xmlns.com/foaf/0.1/"
xmlns:owl="http://www.w3.org/2002/07/owl#"
 xmlns:j.1="http://purl.org/dc/terms/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
 xmlns:j.2="http://purl.org/gem/qualifiers/"
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
 xmlns:j.3="http://online-lab.org/">
 <rdf:Description rdf:about="http://online-lab.org/Agent-creator1">
  <rdf:type rdf:resource="http://online-lab.org/Agent" />
  <j.0:name xml:lang="de">Agent H.S.</j.0:name>
 </rdf:Description>
 <rdf:Description rdf:about="http://online-lab.org/Agent-creator2">
  <rdf:type rdf:resource="http://online-lab.org/Agent" />
  <j.0:name xml:lang="de">Agent C.T.</j.0:name>
 </rdf:Description>
 <rdf:Description rdf:about="http://online-lab.org/Agent-
university1">
  <rdf:type rdf:resource="http://online-lab.org/Agent" />
  <j.0:name xml:lang="de">Technische Universität
Berlin</j.0:name>
  <j.0:name xml:lang="en">Berlin Institute of
Technology</j.0:name>
 </rdf:Description>

 <rdf:Description
  rdf:about="http://www.lila-project.org/interactionpackage-1">
  <j.1:title xml:lang="en">Coupled pendula</j.1:title>
  <j.1:title xml:lang="de">Gekoppelte Pendel</j.1:title>
  <j.1:creator rdf:resource="http://online-lab.org/Agent-creator1" />
  <j.1:creator rdf:resource="http://online-lab.org/Agent-creator2" />

  <j.1:description xml:lang="en">In nature, coupling of vibratory
systems occurs
      frequently. This experiment invetsigates the
      behaviour of two simple pendula that are coupled
      by a spring.</j.1:description>

  <j.1:alternative xml:lang="en">Classical Physics -
Pendulum</j.1:alternative>
  <j.1:language>en</j.1:language>
  <j.1:format>1210x860</j.1:format><!-- this the actual size of the
plugin -->
  <j.1:subject>phyics</j.1:subject>
  <j.1:subject>classical</j.1:subject>
  <j.1:subject>oscillations</j.1:subject>
  <j.1:license>Creative Commons -by-nc-nd</j.1:license>
  <j.1:type>RemoteExperiment</j.1:type>
  <j.2:priceCode>Free</j.2:priceCode>
  <j.3:scientificField>physics</j.3:scientificField>
  <j.3:contact rdf:resource="http://online-lab.org/Agent-creator1" />
  <j.1:rightsHolder rdf:resource="http://online-lab.org/Agent-
creator1" />
  <rdf:type rdf:resource="http://online-lab.org/InteractionPackage" />

 </rdf:Description>
</rdf:RDF>
```


Figure 2: RDF serialization of an interaction package

REFERENCES

[1] iLabs: Internet access to real labs [Online] http://icampus.mit.edu/ilabs/

[2] I. Gustavsson, J. Zackrisson, L. Håkansson, I. Claesson and T. Lagö, “The VISIR project – an Open Source Software for Distributed Online Laboratories”, Proc. of Annual Int. Conf. on Remote Engineering and Virtual Instrumentation, (2007)

[3] S. Khachadorian, H. Scheel, C.Thomsen, P. de Vries, "Deployment of remote experiments: The OnPReX course at the TU Berlin," Education Engineering (EDUCON), 2010 IEEE , vol., no., pp.1065-1070, 14-16 April 2010

[4] S. Jeschke; T. Richter; "Individualization and Flexibility through Computer Algebra Systems in Virtual Laboratories," Multimedia, 2006. ISM'06. Eighth IEEE International Symposium on , vol., no., pp.965-970, Dec. 2006.

[5] D.G. Zutin, M.E. Auer, C. Maier, M. Niederstatter, "Lab2go — A repository to locate educational online laboratories," Education Engineering (EDUCON), 2010 IEEE , vol., no., pp.1741-1746, 14-16 April 2010

[6] D. Lowe, S. Murray, L. Weber, M. de la Villefromoy, “LabShare: Towards a National Approach to Laboratory Sharing” Proc. of 20th Australasian Association for Engineering Education Conference 2009.

[7] eContentplus Project “Library of Labs”, EU grant ECP-2008-EDU-428037, 2009

[8] Application Profile for LiLa [Online] http://www.lila-project.org/

[9] ISO 15836: Information and documentation -- The Dublin Core metadata element set [Online] http://dublincore.org/documents/dcmi-terms/

[10] The Sharable Content Object Reference Model (SCORM) [Online] http://www.adlnet.gov/Technologies/scorm/default.aspx

[11] IEEE Standard for Learning Object Metadata [Online] http://ltsc.ieee.org/wg12/par1484-12-1.html

[12] The Global Online Laboratory Consortium [Online] http://online-lab.org/

[13] DBpedia [Online] http://dbpedia.org/About

[14] Jena – A Semantic Web Framework for Java [Online] http://jena.sourceforge.net/