Standardization Layers for Remote Laboratories as Services and Open Educational Resources

Wissam Halimi[1], Christophe Salzmann[2], Denis Gillet[1], and Hamadou Saliah-Hassane[3]

[1] EPFL, REACT, Station 11, CH-1015 Lausanne, Switzerland
wissam.halimi@epfl.ch, denis.gillet@epfl.ch
[2] EPFL, Automatic Control Laboratory, Station 9, CH-1015 Lausanne, Switzerland
christophe.salzmann@epfl.ch
[3] TELUQ, Université du Québec, Montréal, Canada
hamadou.saliah-hassane@teluq.ca

Abstract. Delivering education and educational resources has evolved from class-centered settings towards distributed, cloud-based models. This is mainly the consequence of publicly available educational resources such as documents, videos, and web applications. At the same time, emerging technologies in information and communication are enabling the development and deployment of remote laboratories on the Web. Today, these freely and openly available educational interactive media are known as Open Education Resources (OERs). Learning management systems, MOOC platforms, and educational social media platforms provide a medium for teachers to create their teaching activities around OERs in a structured way. To enjoy an effective and productive learning experience, it is necessary for the educational resources to be fully integrated in the hosting platform. While most platforms have a ready-to-embed infrastructure for certain types of OERs, they are not ready to host remote laboratories in an integrated fashion. In this paper, we define the necessary integration layers for remote labs in online learning environments. The work is validated by two implementations with different target platforms.

Keywords: lab as a service, remote laboratories, online learning, open educational resources, standardization

1 Introduction

Offering hands-on sessions are one of the main requirements for implementing STEM Education (Science Technology Engineering and Mathematics) [3][8]. By conducting laboratory work, pedagogical objectives such as learning by doing, applying theory to practice, learning to manipulate the physical environment and understanding its flaws and limitations can be attained. Remote experimentation is one way to attain that goal. Broadly speaking, a remote lab is a real physical lab which is accessible at distance through computer networks. More specifically, it is a collection of sensors and actuators, configured to conduct a meaningful scientific experiment, and which can be accessed through a user application over the Internet. In parallel, learning environments

are evolving. Nowadays, there is a shift from classroom based educational settings to distance, blended and other learning modalities which do not constrain the learner in space and time. This is mainly facilitated by the availability of educational resources and web-based educational platforms [10]. UNESCO defines Open Educational Resources (OERs) as being *"any type of educational materials that are in the public domain or introduced with an open license. The nature of these open materials means that anyone can legally and freely copy, use, adapt and re-share them. OERs range from textbooks to curricula, syllabi, lecture notes, assignments, tests, projects, audio, video and animation."* [15]. With the wide online availability of OERs from different sources (Google, educational repositories such as Golabz[4], OER Commons[5] and others), teachers are encouraged to gather relevant resources in a structured format to teach a certain topic or carry out a learning activity such as a lesson in a MOOC (Massive Open Online Course), or an ILS (Inquiry Learning Space) as it will be detailed in Section 3.2.
Generally, an online experiment is conducted separately from pedagogical contexts (lessons). Furthermore, web-based learning environments are not pre-prepared to fully integrate remote laboratories. We recognise that for a remote laboratory to be fully integrated in a platform, it should be able to: (i) retrieve information regarding the context, (ii) provide action logging, and (iii) save and retrieve data. These requirements are essential for an effective educational experience because the user's identity is necessary to be able to link generated experimental data and actions to a specific person in a given context for later awareness and reflection. Moreover, as in a physical hands-on sessions students will be generating data (such as measurements), these data can be exploited by various tools, such as visualization and archiving tools.
Existing remote lab solutions are in the form of standalone applications or web applications. The basic solution for integration in learning environments is to wrap the remote laboratory web interface in an HTML iFrame. This poses a number of challenges to attain the integration goal, and recourse to ad-hoc solutions. The aim of this work is to standardize the design and implementation of pedagogically designed remote laboratories regardless of the target embedding platform.
The paper is structured as follows: in Section 2 we present the proposed standardization layers. In Section 2.1 and 2.2 we present how a remote lab is respectively standardized as an LaaS (Lab as a Service), and how it is later personalised as an OER. In Section 3 we present two use cases of the proposed framework, and we conclude in Section 4.

2 Standardization Layers

Remote laboratories are highly interactive open educational resources. The main goal of remote laboratories is to make available a real physical lab setup at distance. To use a remote lab, users act on the system by sending commands, consequently the lab responds by changing its physical state, and returning its sensor values. Accordingly, remote labs are a wealthy source of different types of data. Through the use of the remote lab, the students produce two types of data: interaction data related to their actions and inputs on the user interface, and experimental data which are the parameters they applied on

[4] http://www.golabz.eu/
[5] https://www.oercommons.org/

the system and the collected results as a consequence. It is unarguably important to make sense out of the activity traces and experimental data which are generated. Therefore, in addition to having both types of data, it is necessary to dispose of the context information. The context is composed of the user identity, the course or the pedagogical scenario in which the resources are utilised and any other details which will contribute to a better perspective of the educational activity.

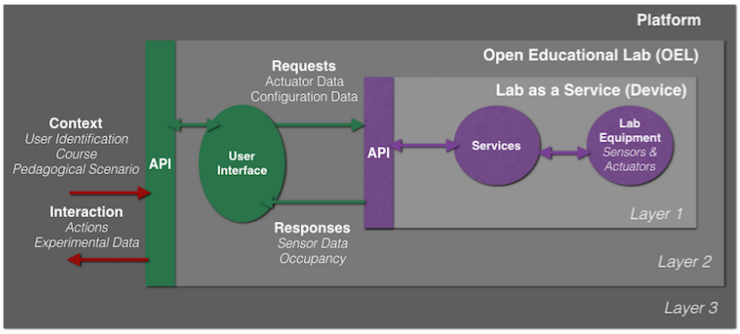

Fig. 1. Standardization Layers for Remote Labs Integration

Figure 1 shows our proposed standardization layers for integrating remote labs in web-based learning environments. This architecture is based on the concept of separation of concerns, where the system is composed of interconnected yet independent components, which communicate through defined interfaces. To this end, our architecture is three-layer and is detailed below.

The first layer (Fig. 1), encompassing the physical equipment of a remote lab is abstracted as a set of software services. This is based on the Smart Device Paradigm that represents a remote lab as a set of services exposed on the Internet through a well-defined API. The Smart Device paradigm enables the independence between the two tiers of the traditional Client-Server architecture adopted for remote labs [11]. Furthermore, it also enables the personalization of the User Interface (referred to as UI and detailed later) in Layer 2. The API provides a set of routines to read and write data from and to the remote lab respectively. It accepts requests for data retrieval from the sensors reflecting the state of the lab, in addition to configuration data that puts the lab in a certain operational mode, if supported. Moreover, there are other requests that the UI can send, for example writing data requests on actuators for controlling the lab. At this level we assume that the lab is capable of accepting requests and sending responses about the sensors and the actuators. The lab as a service is a self-contained layer that is operational regardless of a hosting platform. The information provided by this layer is available to any platform trying to interact with the lab. The concept of LaaS is detailed in Section 2.1.

Next, the remote lab is personalized as an Open Educational Lab (OEL) by the development of a UI integrating the pedagogical elements required by the context, and which

is augmented with the necessary functionalities to insure proper communication with the hosting platform and the interfacing with the remote lab. This is done by calling adequate services of the LaaS. The concept of OEL extending the notion of OER is detailed in Section 2.2.
Last in Layer 3, the OEL is integrated in a hosting platform while insuring the propagation of contextual information, user activity traces, as well as data related to the experimentation itself.

2.1 Lab as a Service

LaaS is a term derived from the XaaS series of terms, where "X" means everything and "aaS" refers to "as a Service". In this paradigm, the assumption is that everything "X" is offered as a service over the Internet rather than at a physical space. It is a notion derived from Service Oriented Computing, where software is made available as a set of services, and hence hiding the dynamics and only exposing the program through a well-described API [4]. "LaaS" refers to Laboratory as a Service, where a laboratory is abstracted and made remotely available through Internet as a software service.
The "Smart Device Paradigm" which aims at separating the two main components which constitute a remote lab architecture: the Client Application and the Lab Server is one way of implementing LaaS. The Client Application is the software interface provided to a user in order to manipulate a remote experiment. The Lab Server responds to the client requests by executing the commands on the physical setup it is controlling. In order to separate the Client Application and the Lab Server, the Lab Server side is equipped with some "intelligence" and follows Service Oriented Computing principles to expose the lab side as software services and hence decoupling the architecture [14][11]. A remote lab standardized as Smart Device provides two types of services: internal and external. The external services are the software interface representing the hardware behaviour of the sensors and actuators that make up a lab. They are the endpoints that allow an outside communication with the physical lab, namely from a client application. They are described through a well-defined and documented API (the *metadata*). The *metadata* is formatted in JSON making it human readable and machine parsable. The *metadata* describing the external services can be used to automatically generate user interfaces [7]. The internal services are suggested mechanisms for lab providers to implement in order to help manage and protect their labs. They are not accessible to the Client Application. Further details on LaaS and the Smart Device paradigm are in [12].

2.2 Open Educational Labs

Remote laboratories are highly interactive educational resources, where the user action has an effect on the system, and which generates data belonging to two categories: interaction data resulting from the use of the UI, and experimental data which are the data sent to the actuators of the remote lab and received from the sensors of the remote lab. Accordingly, collecting data which can be linked back to the user is important for many goals: generated data from the interaction with UI components is valuable for studying interaction patterns, and experimental data are needed by the learners to check their results and possibly use them in other tools. In order to support the full integration of remote labs in target platforms, there is a need to specify the requirements and accordingly develop remote labs as Open Educational Labs. Hence, in this work we consider that access management, activity tracking, and data storage schemes for OELs should be defined to guarantee a full integration of the remote lab in a hosting platform.

Access Management In this work, we are interested in the case where a remote laboratory is part of a complete educational activity (i.e. a lesson). Given that the assumption is that learners connect to the learning activity through an online learning environment, it is usually the case that they dispose of a user identity for authentication with the platform. Referring to Fig. 1, we consider that the remote laboratory will be integrated in the platform through an interfacing module, which in most cases will be a third-party application. To prevent the creation of multiple identities belonging to the same user, it is necessary to propagate the user identity from the platform, to the OEL (Single-Sign On), to the remote lab implemented as a LaaS. More specifically, when learners are conducting their educational activity they should have a unique identity that persists throughout the different sessions and the standardization layers. This guarantees the consistency of reflecting the contexts, saving activity traces, and collecting experimental data. In our proposal, a user authenticates with the platform to get access to the OEL, the OEL authenticates with the LaaS to get access to lab. In Section 3 we provide two examples of remote labs integrated in a LTI consuming platform–edX, and a social media platform–graasp.

User Activity Tracking With the surge of activity tracking in educational settings– today referred to as learning analytics, it is necessary to track a learner's action. The saved information is considered very valuable for many purposes. Using learning analytics, learner success can be predicted, experimental behavior can be mined and understood, moreover adaptation and personalization can be attained [13]. Many authors use learning analytics to understand the behavior of the learner and use it as a feedback for other tools in the platform, such as recommendations [5]. When a remote laboratory is first abstracted as a LaaS, then as an OEL to be integrated in a learning platform, it is clear that there are several sources of activity traces. At the platform level, the log in and log out times would indicate how much time a learner spend in the lesson for example. At the LaaS level, keeping records of the different exchanged requests and responses with the UI can help in bringing meaningful insight into lab usage. Precisely, the experimental parameters can be used to extract use patterns for a certain experiment and hence understanding how students are using the lab when studying a certain concept. In our approach, since we want to support a consistent identity of a user throughout the layers, we believe the most adequate solution is to have a common repository for actions coming from the different layers, where a user's actions are identified by the identity coming from the hosting platform. The common repository could be proprietary to the platform or external to it as it will be shown in Section 3.

Data storage and retrieval mechanisms When conducting an experiment, students generate the results of applying parameters on the process of the given lab. And just like in physical hands-on sessions, the data need to be collected and archived for future use, such as for graphing or tabulating results. Moreover, considering that the remote lab is embedded in a learning platform in the context of a lesson, additional tools consuming the data could be added for pedagogical purposes as it will be shown in Section 3.2. To improve user experience, it is necessary to specify mechanisms for data saving and retrieval. As for the repository of learning activity tracks, the database keeping the experimental data can be specific to the platform or external. But in both cases, the consistent user identity through the standardization layers should be used as an identifier for the data.

2.3 Integration in a hosting platform

Once a laboratory setup is abstracted from the physical world as a set of services based on the Smart Device paradigm, it is ready to be personalised for use. This is done by building an application which invokes the API calls of the LaaS to gain access to it. At this stage, the remote laboratory can be exploited without any context (i.e. without being part of a lesson). But if chosen to be used in a pedagogical scenario, augmenting the LaaS with a UI (as a front-end), and with user identity management, activity tracking, and experimental data management (as back-end) turns it into an OEL ready to be integrated in a hosting platform. Needless to say that the interfaces used with the hosting platform for the mentioned requirements will be specific and cannot be standardized. This is the level where the user credentials are managed, the activities traces are consumed, and the experimental data is saved. In the next section, we present two implementations of two different remote labs integrated in two online learning environments with different infrastructures.

3 Use Cases

In this section, we present two examples of remote laboratories developed and integrated in learning environments as per the proposed guidelines. We will first present the example of a control system lab which is integrated in edX[6]. Then we will detail the Mach-Zehnder interferometer remote lab which is integrated in an educational social media platform: graasp[7].

3.1 MOOLs for MOOCs

The lab we are considering for this example is a control systems lab designed and implemented to service a large number of users. It is integrated as part of a control systems course, designed and deployed on a local copy of edX[8] for EPFL. The complete infrastructure of the lab is made of multiple replicas of the same lab setup serviced on the Internet by Smart Devices, an HTML UI to be integrated in edX, a *.cgi* interface for LTI authentication, database and other services, and an edX server.

[6] http://www.edx.org
[7] http://www.graasp.eu
[8] http://www.edx.epfl.ch

Integration in edX using existing standards LTI (Learning Tools Interoperability) is a specification developed with the principal goal of standardizing the integration of rich, third-party learning applications with educational environments such as learning management systems, portals, learning object repositories, or others. When talking about LTI, the learning applications are called Tools (hosted by Tool Providers–TP) and the integrating platforms are called Tool Consumers (TC). The main outcome of implementing the LTI specification is enabling the seamless integration of remotely hosted third-party content in a given online learning platform, while communicating the user identity and context to the tool, without any ad-hoc solutions [1]. In our context, a Tool is the OEL, and the Tool Consumer is the integrating learning platform edX.

The LTI implementation for this lab contains the following required parameters: *lti_message_type*, *lti_version*, and *resource_link_id*. In addition to the following recommended parameters: *user_id*, *roles*, and *context_id*. More information on the meaning of these parameters can be found in [1].

Given that the course structure was customized to group each learning activity (lesson) with its corresponding resources, there was a need other personalisation parameters. Hence the LTI specification was extended to include the following fields:

- *experiemental parameters*: which will allow to invoke commands on the lab once the LTI authenticated with the remote lab.
- *experiment identifier*: since in a single page characterized by a *context_id* and containing a lesson, we can have multiple tabs in which there are remote lab UIs, there is a need to identify each tab for activity tracking and data archiving.
- *experiment duration*: the teacher can set the time of the experiment through edX in the LTI parameters. This will limit the amount of time students can spend doing an experiment if others are waiting for turns.

At the time of the integration of the remote lab, edX didn't implement an LTI version which supports saving and retrieving data to the platform. Hence, a *cgi* interface was put in place between edX and the external tools. The *cgi* interface validates the LTI encoded request containing the edX user ID and other context related information. Once the request is validated, the LTI module content is integrated as an iFrame in the edX page (Fig. 2).

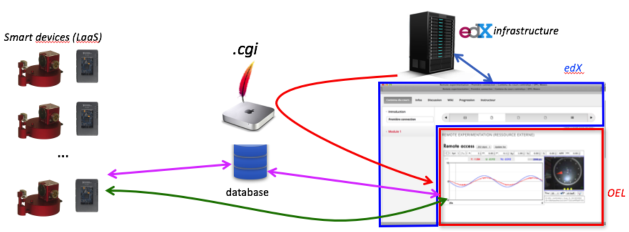

Fig. 2. OEL Integration in edX

3.2 The Mach-Zhender Interferometer

This lab example is an interferometer to study light interference at high school. It is implemented in order to be integrated in a Learning Inquiry Space (ILS) which introduces the phenomena of light interference. In the context of the Go-Lab project, an ILS is a tool embedding a pedagogical structure and resources to complete an inquiry learning activity [6]. Concretely, the learning activity is divided into five basic phases, through which students learn about science the way scientist do it. Teachers build their ILS in Graasp by embedding different educational resources, including remote labs [9].

The physical lab is made out of many sensors (e.g. photodiode) and actuators (e.g. piezo controller) configured to experiment with several properties of light diffraction and interference; the details of which are not of interest for the purpose of this paper, for further information refer to [7]. The lab is abstracted as a set of services and exposed through an API. A Smart Device implemented on an embedded computer (myRIO[9]) interfaces the hardware and handles user clients requests and responses.

Integration in Graasp with an ad-hoc solution Graasp– the hosting platform, provides a mechanism for integrating third-party applications enabling them to use its proprietary services (context information, user identity, activity tracking, saving and retrieving data). This is done by putting in place an OpenSocial container which plays a proxy between graasp's API and third-party applications [2], in addition to the ILS library[10] which takes care of ILS specific mechanisms. Fig. 3 shows the integration of the OEL in Graasp, within a learning activity. Accordingly, an OpenSocial application which provides a UI to control and observe the lab is implemented and integrated in Graasp. In addition to communicating with the Smart Device, the OpenSocial application is aware of the user identity through the *People API* and saves associated activity tracks (through the *ActivityStreams* API) and experimental data (through the *Documents API*).

When in an ILS, students start sequentially with the *Orientation*, *Conceptualization* phases which respectively introduce the subject and ask the students to hypothesis about it. Usually the practical work to validate or refute the hypothesis is done in *Investigation*. The implemented OEL saves the interaction traces with the UI in compatible formats of the platform[11], to be used later by other apps. The experimental data is also saved in adequate formats to be used by other tools, such as the Data Viewer[12] tool. In the *Conclusion* phase the students conclude about their experimentation results and hypothesis, then in *Discussion* they can share with their instructor and peers their findings.

4 Conclusion & future work

In this paper we presented our standardization architecture for integrating remote labs in online learning environments. Our approach is driven by the need to support the

[9] http://www.ni.com/myrio/
[10] https://github.com/go-lab/ils
[11] https://github.com/go-lab/ils/wiki/ActionLogger
[12] http://www.golabz.eu/apps/data-viewer

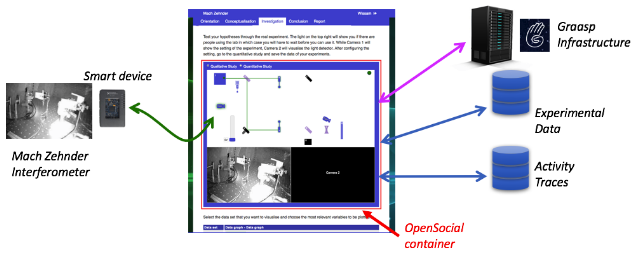

Fig. 3. OEL integration in Graasp

development and deployment of remote labs which are easily integrated in different educational platforms. At a first level, a remote lab is abstracted as a set of web services accessible through APIs. This allows the personalisation of the OEL with a UI augmented with full-integration requirements (context awareness, activity tracking, and experimental data storage) becomes an Open Educational Lab. The resulting OEL is expected to interoperate with the different services and other Open Educational Resources used in a learning scenario, hence providing an online learner what is a good experience. We later present two remote laboratories integrated in two different online platforms, with pedagogically sound resources and interaction features. The control system lab is integrated in an LTI consuming platform. The lab is integrated as an OEL by implementing the communication with the LTI container of the platform (single-sign on for both platform and lab) and extending its properties to support further user needs (data saving and retrieval). The other example, is an interferometer integrated in an educational social media platform which supports the requirements for integration. The lab is integrated as an OEL by using the services provided by Graasp for user identity, activity tracking, and data storage and retrieval. It is worth mentioning that the proposed solution for edX could also reused with little effort in other environments supporting LTI such as Moodle.

Acknowledgment. This research was partially funded by the European Union in the context of Go-Lab (grant no. 317601) projects under the ICT theme of the 7th Framework Programme for R&D (FP7). This paper is one of the contributions of the task carried on in the framework the IEEE-SA P1876^{TM} Working Group on “Standard for Networked Smart Learning Objects for Online Laboratories”[13].

References

1. Learning Tools Interoperability - Implementation Guide. `https://www.imsglobal.org/specs/ltiv1p2/implementation-guide` (2016), [Online; accessed 15-January-2017]
2. Bogdanov, E., Limpens, F., Li, N., El Helou, S., Salzmann, C., Gillet, D.: A social media platform in higher education. In: Global engineering education conference (educon), 2012 ieee. pp. 1–8. IEEE (2012)

[13] https://ieee-sa.imeetcentral.com/1876public/

3. Dasarathy, B., Sullivan, K., Schmidt, D.C., Fisher, D.H., Porter, A.: The past, present, and future of moocs and their relevance to software engineering. FOSE 2014 Proceedings of the on Future of Software Engineering pp. 212–224 (2014)
4. Duan, Y., Fu, G., Zhou, N., Sun, X., Narendra, N.C., Hu, B.: Everything as a service (xaas) on the cloud: origins, current and future trends. In: 2015 IEEE 8th International Conference on Cloud Computing. pp. 621–628. IEEE (2015)
5. Duval, E.: Attention please!: learning analytics for visualization and recommendation. In: Proceedings of the 1st International Conference on Learning Analytics and Knowledge. pp. 9–17. ACM (2011)
6. Gillet, D., De Jong, T., Sotirou, S., Salzmann, C.: Personalised learning spaces and federated online labs for stem education at school. In: Global Engineering Education Conference (EDUCON), 2013 IEEE. pp. 769–773. IEEE (2013)
7. Halimi, W., Jamkojian, H., Salzmann, C., Gillet, D.: Enabling the automatic generation of user interfaces for remote laboratories. In: Remote engineering and virtual instrumentation (rev), 2017 14th international conference on. IEEE (2017)
8. Lowe, D.: Mools: Massive open online laboratories: An analysis of scale and feasibility. 2014 11th International Conference on Remote Engineering and Virtual Instrumentation (REV) pp. 1–6 (2014)
9. Rodríguez-Triana, M.J., Govaerts, S., Halimi, W., Holzer, A., Salzmann, C., Vozniuk, A., de Jong, T., Sotirou, S., Gillet, D.: Rich open educational resources for personal and inquiry learning: Agile creation, sharing and reuse in educational social media platforms. In: Web and Open Access to Learning (ICWOAL), 2014 International Conference on. pp. 1–6. IEEE (2014)
10. Saliah-Hassane, H., Reuzeau, A.: Mobile open online laboratories: A way towards connectionist massive online laboratories with x-api (c-mools). In: 2014 IEEE Frontiers in Education Conference (FIE) Proceedings. pp. 1–7. IEEE (2014)
11. Salzmann, C., Govaerts, S., Halimi, W., Gillet, D.: The smart device specification for remote labs. In: Remote engineering and virtual instrumentation, 2015 12th international conference on. pp. 199–208. IEEE (2015)
12. Salzmann, C., Halimi, W., Gillet, D., Govaerts, S.: Deploying large scale online labs with smart devices. Cyber-Physical Laboratories in Engineering and Science Education (under review) (2017)
13. Siemens, G., d Baker, R.S.: Learning analytics and educational data mining: towards communication and collaboration. In: Proceedings of the 2nd international conference on learning analytics and knowledge. pp. 252–254. ACM (2012)
14. Tawfik, M., Salzmann, C., Gillet, D., Lowe, D., Saliah-Hassane, H., Sancristobal, E., Castro, M.: Laboratory as a service (laas): A model for developing and implementing remote laboratories as modular components. In: Remote Engineering and Virtual Instrumentation (REV), 2014 11th International Conference on. pp. 11–20. IEEE (2014)
15. UNESCO: What are Open Educational Resources (OER). `http://www.unesco.org/new/en/communication-and-information/access-to-knowledge/open-educational-resources/what-are-open-educational-resources-oers/` (2016), [Online; accessed 30-December-2016]