Using e-Learning Standards to Improve Serious Game Deployment and Evaluation

Pérez-Colado, Iván José
Dept. of Software Engineering and Artificial Intelligence
Complutense University of Madrid
Madrid, Spain
ivanjper@ucm.es

Pérez-Colado, Victor Manuel
Dept. of Software Engineering and Artificial Intelligence
Complutense University of Madrid
Madrid, Spain
victormp@ucm.es

Martinez-Ortiz, Iván
Dept. of Software Engineering and Artificial Intelligence
Complutense University of Madrid
Madrid, Spain
imartinez@fdi.ucm.es

Freire, Manuel
Dept. of Software Engineering and Artificial Intelligence
Complutense University of Madrid
Madrid, Spain
manuel.freire@fdi.ucm.es

Fernandez-Manjón, Baltasar
Dept. of Software Engineering and Artificial Intelligence
Complutense University of Madrid
Madrid, Spain
balta@fdi.ucm.es

***Abstract*— The serious game industry has been a subject of interest for the educational community in the last few years. However, despite their potential benefits when compared to traditional e-learning, use of serious games in education has been hampered by difficulties, including that of deploying games in multi-platform settings and leveraging data generated by their players to improve the learning outcomes of those players. This increasing uptake requires to simplify serious games application for both teachers and developers. This paper proposes a standards-based approach to improve serious games deployment and evaluation. Building on existing and upcoming standards will allow games to be seamlessly integrated into the learning management systems that are already present in most higher education institutions. We present initial experiments that put the proposed approach into practice using two specific tools: uAdventure and SIMVA.**

Keywords—serious games, e-learning standards, learning analytics

I. INTRODUCTION

In recent years there has been a great interest in the use of serious games (SG) in education. SG are an effective and highly motivational tool that can be applied not only in learning or training activities [1], but also in an educational context to change the players' attitude (e.g., bullying [2]).

Two of the most relevant aspects for teachers about SGs are the effectiveness of the games and the ease of deploying them in a classroom. The former is relevant to the teachers in order to apply SGs effectively not only as a motivational tool, but as an educational tool / activity integrated as part of the curriculum. The latter is also relevant to the teachers, especially if it is used as a proper educational tool, where the allotted timeslot for playing the game is limited and so it is required to smoothly change from educational activities in an easy way. In addition, teachers would like to have more control and supervision about the game-based activity. That will require a tool to monitor the students' progress through the activity, so the teacher can intervene during the session (if it is required) or simply to better understand what student actually do (student assessment).

Those previous aspects are related, because of during the development cycle of the SG it is also desirable to deploy the SG in an education setting in order to evaluate its impact and improve it (if necessary).

Traditionally, games in education are considered as black boxes [3], [4], because from outside of the game it is not usually possible to know how the player is performing. In the best of the cases, games just provide minimal feedback or a single student grade [5]. However, nowadays it is more common to find out SGs that apply learning analytics to capture players' gameplay interactions in order to help to the teachers to understand the students' progress through the game and if they actually learnt [6]. But SG learning analytics requires at least a repository to store the analytics and usually a tool that analyze the data and provide a dashboard for the stakeholders (e.g., teacher). Moreover, SG deployment usually require additional effort from teachers (e.g. requiring to setting up accounts for their students, setting up activities, etc.) and this could be done using a different tools outside from their current virtual campus.

We believe that the identified issues share common technical difficulties that can be addressed by applying existing technical e-learning standards and specifications.

The rest of the paper is structured as follows: Section II analyzes problems encountered by teachers that tryto use SGs as another educational resource, as motivating element of our work. Section III analyzes the most relevant e-learning technical standards and specifications that can help to address the previously-identified issues from a technical perspective. Section IV proposes a standards-based approach to improve game deployment and evaluation, and Section V describes initial experiments of our standards-based proposal using uAdventure and SIMVA. Finally, section VI contains conclusions, describes limitations to our approach, and outlines future lines of work.

978-1-6654-4434-7/22/$31.00 ©2022 IEEE
XXX-X-XXXX-XXXX-X/XX/$XX.00©20XX IEEE

II. Using game-based learning in education

Teachers that want to include game-based learning in Learning Management System (LMS) courses must face multiple challenges, as described in [7], [8]. The main challenges they must address are:

1) Finding a suitable game.
2) Providing the game to students.
3) Integrating the SG as an activity inside a course.

In addition to recommendations from other teachers, they discover the SG by using general-purpose search engines, or through apps stores. However, it is quite common for teachers, especially for non-university studies, to have access to specific educational resources repositories provided by associations such as European Schoolnet (http://lre.eun.org/), provided at national level such as the Agrega2 initiative in Spain (http://agrega.educacion.es/) or created by a community of practice such as the EDIA Project in Spain (https://cedec.intef.es/proyecto-edia/). Usually, these specific-purpose platforms provide search capabilities that not only allows the teacher to search by keywords, but also to refine the search using other aspects such as educational level, technical aspects (e.g. compatible platforms), a predefined taxonomy for a common curricula, etc.

Once the game has been found, teachers must provide the game to the students. This process depends heavily on the SG technology, that is the technology in which the game has been developed. In the best-case scenario, the SG is developed as a web-based game, so the students just need to have access to a browser. But, even in this case, the teacher still may require to setting up accounts for each of the students and give them access to the game. However, it is quite common to create SGs as native applications (smartphone, tablets or desktop) in order to take full advantage of the device's technical capabilities and overcome the current technical limitations of web-based content. In this scenario, teachers may need to install and configure the SG in all the school's devices or, at least, provide guidance and support to install and play the SG on the students' devices (if BYOD approach is used).

To perform the tasks of configuring and deploying the games, the teacher may have a tool to simplify the work. But even in that case it is an extra effort as those tools are not usually well integrated into a school's Learning Management System (LMS). Besides, if the SG is used as means to evaluate or grade the student, an additional effort is required to capture/write down/export the evidences generated by the SG per student and import them into the LMS as part of the students' records or portfolio. In case that the SG that include learning analytics the configuration could be more complex as it could require additional data (e.g., configuration of the Learning Record Store). Moreover, the privacy and data-management policy of the schools, may complicate even more the process, requiring even to disable the learning analytics if it is not possible to store the data in a specific platform, such as the organizational LMS.

Other option is that teachers are the SG creators. Although teachers can develop SGs themselves by using both a game engine/platform such as Unity or an authoring tool such as *uAdventure* [9], we do not consider it feasible for the average teacher, due to the effort and knowledge required to carry out the tasks. So, we consider that is more realistic that the teacher uses an authoring tool to adapt a pre-existing SG to their own needs and in this case the tool should help them to address the identified issues. In case of a developer that create a SG from scratch, it should take those deployment issues into account to facilitate the SG use in real settings. In both cases, creating an ad-hoc solution for the identified challenges are expensive to develop and maintain, and may lead to vendor lock-in.

III. e-learning Standards used by Open Educational Resources and tools

Some of the identified issues are similar to the problems that Learning Objects (LOs) [10] have already addressed in the past. Moreover, we can assimilate, to some extent, a serious game as a learning object, or at least an educational activity, hence we can take advantage of the previous experiences that addressed interoperability and deployment issues of LOs. Those issues led to the development of several technical specifications and standards aiming to facilitate the adoption and usage of LOs and to facilitate the interoperability of educational tools provided by different vendors.

As part of the Learning Objects ecosystem, several specifications and standards have emerged in order to provide interoperability. These different standards had tried to address different aspects of the LO interoperability. For example, the packaging and distribution, tagging and discoverability, organization and hierarchical structure of LOs and learning activities. Those standards also cover other aspects such as communication, sequencing and tool interoperability [11]. For teachers and educational managers, standards offer a guarantee to rely on when integrating content in their courses.

The following subsections briefly introduce the most relevant standards that we consider that help to address the identified issues in the previous section and analyzed them through the scope of their application in serious games.

A. Discovering serious games

IEEE Learning Object Metadata (LOM) is an educational content tagging specification [12] that is used to describe LOs. The metadata generated using IEEE LOM and associated to a LO can be used to understand the nature of the content, classify and organize it. One of the common applications of IEEE LOM-generated metadata accompanying an LO is to facilitate the indexing and searching of LOs in a repository [13], [14] [15]. Due to its flexibility and widespread adoption in different tools and context, it is also used as part of other standards such as Shareable Content Object Reference Model (SCORM) and IMS Content Packaging (IMS CP) [16].

Besides IEEE LOM, there is a new family of standards ISO/IEC 19788 Metadata for Learning Resources [17] that want to address some of the limitations of IEEE LOM. Despite that it is still evolving and maintained, however its adoption is not as widespread as IEEE LOM.

B. Accessing activities

IMS CP is a content packaging standard [16]. It offers a multi-level content structure model using a single manifest file packaged along the contents within a zip file. This model has

been widely adopted by the most popular LMSs and is used for game packaging in learning environments.

IMS CP allows an LMS to distribute web-based content – anything that the LMS can handle, CP can package and, can include LOM metadata. Packaged contents can be extracted to be reused separately. When packaging games or other highly interactive content, IMS CP allows web-based distribution for web-based technologies. In the past, this was suitable for Java or Flash games; but in newer browsers, only pure web-based technologies are currently available, such as HTML5 and WebGL. Due to the self-contained nature of the IMS CP packages, it forces the LMS to directly deliver content to users with two main drawbacks. First, games can include heavy multimedia resources, placing a significant load on the institutional infrastructure (usually a LMS). Second, there is tight coupling between the game and the course where the game is used. Updating or improving the game requires authoring access to the course where that game is deployed.

The IMS Learning Tools Interoperability standard focuses on communications between LMS platforms and tools [18]. It provides a mechanism to forward user authentication and authorization from the LMS to an external tool for the duration of a session. During the session the tool can report backs some information such as completion and grades for tasks carried out by students. LTI can be launched in multiple ways, including within a new browser window, or embedded inside the LMS. However, lately some browsers have modified cookie access in external tools launched inside of a window, limiting certain use-cases. Additionally, LTI Submission Review lets the tool leave a link to a page that will let the teacher access to additional information about the activity.

On the usability side, LTI is simple to use for teachers, which only need to provide a link to the LMS that can then automatically configure the corresponding activity. Nevertheless, on the developer side, it is a complex standard to implement and because of its cookie-based nature it cannot handle native applications. LTI is currently adopted by most major LMSs, although some do not yet support the latest version (currently v1.3).

Finally, IMS Common Cartridge (CC) is a IMS specification similar to SCORM, that also uses IMS CP as a packaging base, but with a more flexible approach [18]. It provides better mechanisms to build packages from modules, greatly facilitating reuse of content within a package. It can also include LTI links as content, simplifying web-distribution and communication activity with LMS (see next section).

C. Communication

Certain standards bridge several categories; for example, we have listed IMS LTI under “access”, since it can be used to launch activities. It can also, in terms of communication, inform activities of who their users are from the LMS’s point of view, and allow activities to access the LMS’s API for grading and completion purposes.

The Shareable Content Object Reference Model (SCORM) is a collection of standards and specifications for Learning Objects developed by ADL. It builds on the IMS CP standard for packaging and IMS LOM standard for tagging with specific profiles; and then adds a simple communication layer. It can even support sequencing, by using the IMS Simple Sequencing (SS) standard. There have been several widely adopted versions of SCORM, with the 1.2 and 2004 versions being the most popular. SCORM has been used for game-based learning, but presents several problems in addition to those it inherits from usage of IMS CP [19]. The first issue is the constrained information stream it provides, often used just to report a result-score pair. This black-box approach is a severe limitation with highly interactive contents such as games, which could report a much richer feedback. Additionally, SCORM only works for web-based platforms, and makes no provision for other platforms. Since web-based gaming is not mainstream, and both iOS and Android have only partial support for WebGL, this is a severe drawback for deploying modern games. It is possible to use a proxy approach to avoid this web-based constraint (such as the ones provided by Rustici Software). However, since these solutions are still based on browsers, they can be fragile over time. Finally, SCORM is also insecure and easily hackable [20].

The xAPI specification provides both data model and an Application Program Interface (API) to work with Learning Analytics [18]. Its focus is to provide a flexible format to trace all meaningful interactions, using simple statements that include an actor, an object and a verb, and can optionally include results, score and a set of customizable extensions. The xAPI model can be extended with profiles that add more verbs, objects and extensions to describe specific domains. For games, the Serious Game xAPI profile adds a set of verbs including the concept of stages, dialogs, choices, interactive elements and meaningful variables [6]. In contrast to other specifications, this is just a data formatting specification, not a communication protocol. Thus, the xAPI model cannot be used by itself and requires an additional communication protocol to initiate, report on, and end an xAPI activity stream. The xAPI specification comes along with the Learning Record Store (LRS) system specification, intended to store xAPI traces and allow the LMS or other tools query and retrieve them [18].

IMS Caliper is a related standard that also supports activity streams, using events instead of statements, each with an actor, action and object which align with xAPI’s actor, verb and object. These events can also use specific vocabularies. However, adoption of Caliper is currently lagging that of xAPI.

In recent years, interest in activity streams such as those generated by xAPI and Caliper has experienced a steady growth, since those streams enable much better analytics for different stakeholders. From the perspective of learner and teachers, activity streams provide rich reporting, allowing teachers to better understand aspects such as interest or knowledge; and to detect students that become stuck. This, however, requires a suitable analysis layer. Researchers can use activity streams to complement traditional validation systems and extract more conclusions through machine learning. Finally, developers can also find them useful to identify possible design flaws and bugs.

The ADL and AICC CMI-5 specification that address both the packaging and deployment of simple or structured educational resources [18]. The CMI-5 uses xAPI and defines an specific application profile, to help with the coordination between the LO (known as assignable units in the context of the

specification) and the LMS, but it is also possible to use other profile to provide specific and detailed analytics as part of the user interaction with the LO.

CMI-5 can be considered the spiritual successor of SCORM, addressing some of its weaknesses. CMI-5 packages include an xml manifest that describes the elements in the course, including Assignable Units (AUs) which represent xAPI activities, and nestable Blocks to provide a hierarchical structure. CMI-5 uses a specific xAPI profile to communicate with the LRS and describe the state of activities, using extensions for its outcomes.

One of the most important aspects of CMI-5 and its application to serious gaming is that the AU launching protocol is based on a URL redirection. This mechanism it is usually used to redirect the user between web servers. However all major operative systems (e.g. Windows, MacOS, most Linux distributions) and mobile platforms support the definition of custom schemes (e.g. http://, https://, myapp://) that allows to seamless transition from a web context to a native application. Using this approach, when launching activities, LMSs can provide the authorization and user information as the actor to the corresponding application, solving the task of identifying the user and gathering the user information in the activity and thus simplifying the teacher experience. While CMI-5 was launched in 2016, it remains unsupported by the main LMSs.

TABLE I. E-LEARNING STANDARDS AND THEIR PURPOSES

	Feature					
	Tagging	***Packaging***	***Delivery***	***Protocol***	***Report***	***Data Model***
LOM	X	-	-	-	-	-
CP	LOM	X	-	-	-	-
SCORM	LOM	CP	-	X	X	-
LTI	-	-	X	-	X	-
CC	LOM	X	LTI	-	LTI	-
xAPI	-	-	-	-	-	X
CMI-5	Basic	X	X	X	X	X

IV. A STANDARDS-BASED PROPOSAL FOR SERIOUS GAMES DEPLOYMENT

Based on the identified standards and specifications described in the previous section, this section describes how they can be combined to facilitate the deployment of serious games in educational settings. We can distinguish two main use-cases: that of an LMS-hosted serious game; and that of a native serious game that we wish to integrate into an LMS. In both cases, we wish to minimize the work for developers, teachers and students for deployment and launch; while also gathering rich interaction via learning analytics, and reporting results back to the hosting LMS.

A. In-LMS games using WebGL and IMS CP

Our first proposal (Fig. 1) focuses on the problem of delivering games aiming for simplicity and compatibility; with optional grading and evaluation. We have chosen IMS CP for packaging because it is widely supported and can run games using web-based technologies, in particular WebGL, the only web technology with significant adoption after the demise of Java and Flash. For developers thinking of this solution, games can be implemented using WebGL-native engines such as Babylon.js, PlayCanvas or Three.js; or popular cross-platform engines such as Unity or Unreal Engine, which provide a degree of future-proofing as platforms continue to evolve. To include the game in the courses, the game must be then compressed together with the imsmanifest.xml file describing its entry point; and uploaded to the LMS, which will deliver it to the web browsers of players.

Additionally, to solve the problem of integrating the game experience with course grading and evaluation, the game can support the SCORM communication protocol. However, we recommend including an xAPI trace tracker with an anonymous PIN-based authentication that will both avoid SCORM's the communication limits and simplify later CMI-5 support. With such a tracker, the WebGL game would send xAPI traces to a pre-configured external LRS for later review.

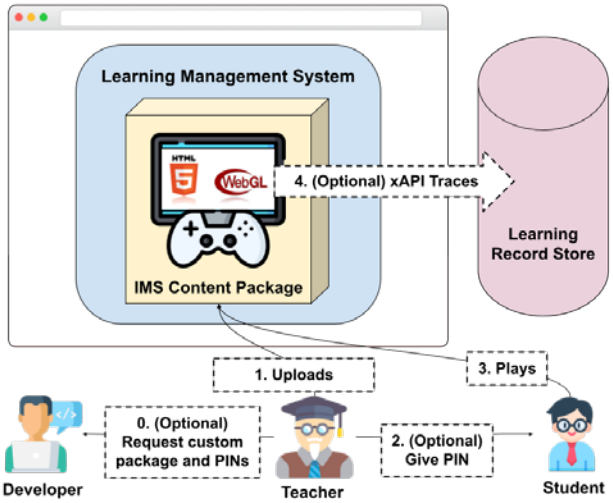

Fig. 1. Using games in the class using only the LMS and Web-Based technologies with optional pin authenticated xAPI-Based Analytics.

Currently, this approach can be applied in a wide variety of learning platforms, as both IMS CP and SCORM are specifications that are widely supported. In particular, they are supported in the "big five" LMSs: Moodle, Blackboard, Canvas, Desire2Learn and Sakai.

B. Externally-hosted games, launched from the LMS via LTI.

Our second proposal (Fig. 2) focuses on the task of delivering games using standards that go beyond the limits of IMS CP and SCORM, providing support for native applications, better game distribution, and seamlessly integrating grading, evaluation, and rich analytics with xAPI.

CMI-5 specification is the xAPI companion (https://adlnet.gov/projects/cmi5-specification/), because it solves the problem of how to establish the communication between a specific activity or educational resource that generates xAPI statements and an LRS that stores them, since xAPI only describes the protocol to interact between them, but not how to establish this initial connection. Although it is expected to be widely supported in the future years, as of this writing it is not yet supported in any of the main LMS. For this

reason, we have considered another approach for hosting xAPI-based CMI-5 packages. This approach uses the widely supported LTI standard to delegate the CMI-5 package launch to an intermediate xAPI-compliant tool that also contains an LRS to store the traces. Rustici Software has implemented a similar solution in Rustici Dispach.

With LTI, the LMS creates an identified bidirectional communication pipe with the mediator tool. In one direction, it delivers the user's auth session to the mediator platform, and in the other direction it reports results from the activity back to the LMS. When the activity finishes, the teacher will be able to access the xAPI reports through LTI Submission Review.

User experience is also an important factor in this approach. The students will have a seamless experience with a one-click launch of the game by following a link presented by their LMS (the CMI-5 launch is seen as a redirection), giving the developers greater freedom to choose where the game is hosted, and both boosting the speed with which the game can be launched and allowing easier access to patch possible game issues. Teachers can also enjoy the benefits: when game packages are managed by developers, teachers just have to include the appropriate LTI links in their LMS courses; otherwise, teachers would first have to upload the packages to the mediator tool to obtain the corresponding LTI links.

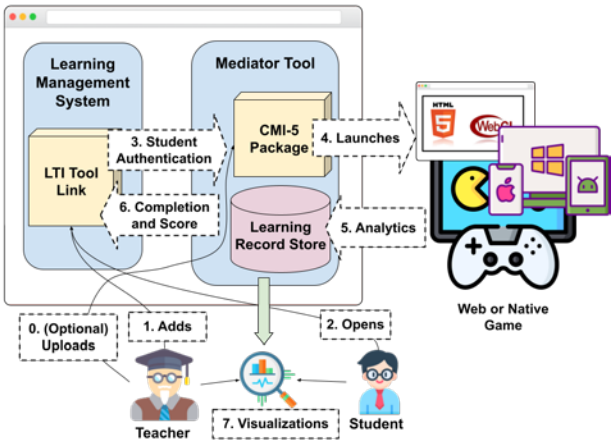

Fig. 2. Using games in the class with LTI and CMI-5 through a mediator tool that supports CMI-5 Launch for xAPI-Based analytics.

IMS LTI is one of the specifications with one of the highest adoption rates among LMS and other educational tools. At the time of writing this article (February 2022) there are 201 certified tools and platforms that comply with this spec (see https://site.imsglobal.org/certifications). Among them, again we can find out the *big five* LMSs.

V. Initial Experiments

We have implemented both proposals as a working proof of concept. Our implementation uses two of our main products: *uAdventure* is used for game developing, packaging and xAPI support; and *SIMVA* provides an LTI connection, CMI-5 inspired launch and LRS-like functionality.

uAdventure is a serious game authoring tool for point-and-click adventures created on top of the cross-platform Unity engine with both native and Web-GL support [21]; and is useful for non-experts since it requires no programming knowledge. *uAdventure* supports Learning Analytics by implements xAPI out of the box with a tracker that can send traces to an LRS.

On the *uAdventure* side of the proposal implementation we included both packaging and data communication. Packaging is performed using a semi-automated process that generates IMS CP packages with WebGL builds. Since IMS CP supports IMS LOM for tagging, we have also provided a metadata editor that fills many of the required fields from the game descriptor. As for the data communication, the tracker can be connected to an LRS using a configuration file that must also reference single sign-on to obtain the necessary authorization. Once *uAdventure* obtains the session authorization via OAuth 2.0 or OpenID, it can include actor information in each trace. In addition, to support non-web games, *uAdventure* games can be launched using a multi-platform URI Scheme.

Until this point, the presented implementation is enough to test that the first proposal is indeed practical. The IMS CP packages exported have been successfully tested in Moodle from both desktop and Android platforms. Although Unity WebGL works in mobile it is not officially supported. Regarding communication support, The IMS CP package can also send xAPI traces using the *uAdventure* tracker, but to better understand this we must first introduce *SIMVA*, which receives the traces sent by the *uAdventure* games.

SIMVA stands for Simple Validator, and is a multi-purpose validation tool that includes user management, experiment design, cloud storage, and data-science [22]. In terms of user management, it is GDPR compliant, focused on anonymity, implementing a pseudonymous PIN-based system to identify user traces. *uAdventure* includes a simple *SIMVA* wizard that can create an activity for a game and generate a PIN-code for each expected participant. From the student and teacher perspective, this is relevant once the game is opened, as after the LMS loads the IMS CP contents, the game will request a PIN to send the traces to *SIMVA*.

To test the second proposal, we use *SIMVA* as a mediator tool. This requires *SIMVA* to: i) provide interoperability with an LMS through LTI; and ii) host IMS CP or CMI-5 packages and provide the necessary user authorization to communication with them (currently ongoing). We have already implemented full LTI support for *SIMVA* to be used as an external tool from the point of view of the LMS; while also acting as a host platform that can serve games. This allows *SIMVA* to be used not only as part of an external LMS-hosted course, but also, in the future, to host courses itself and point to external tools. We are currently working on hosting, having achieved basic IMS CP support. We are therefore making progress towards extending this support to encompass CMI-5 in both *uAdventure* and *SIMVA*.

VI. Conclusions and Future Work

Game based learning offers educators significant benefits compared to traditional methods. To make the best of those benefits, serious games should be easy to find, distribute and integrate. Using standards for these tasks is important to guarantee the reliability and interoperability. However, approaches made in the past such as SCORM have demonstrated limited capabilities, limiting games to web-based, black-box

activities. In contrast, use of more recent standards such as xAPI can greatly increase collection of data with which to perform Learning Analytics; while standards such as CMI-5 no longer limit the platforms on which games can be played.

The standards we discussed show very diverse capabilities in both game access and data communication and uncover a lack of direct support in xAPI-related standards in the main LMSs (Blackboard, Canvas, D2L, Moodle and Sakai). For this reason, we propose two different approaches to integrate games in LMS courses, both without and with xAPI-based Learning Analytics. The first approach relies on the IMS Content Package standard using WebGL, offering both cross-platform support and tailored xAPI analytics support using open source tools. The second approach uses the IMS Learning Tools Interoperability standard to integrate a mediator tool that supports launching xAPI activities, which should be packaged using CMI-5. This "proxy" approach can launch both online and native games and transparently handle player authorization during the activity. However, this is more complex and requires more infrastructure to setup.

We have tested implementations of both approaches by extending two existing tools, *uAdventure* and *SIMVA*. Starting with the first approach, with *uAdventure*, the teacher can edit serious games, export them as WebGL, and package them as IMS CP; and use the out-of-the-box Learning Analytics support to send traces to a previously configured LRS. To identify the users, we have tested a PIN-based system, where pseudo-random PINs are given to students to guarantee their anonymity while ensuring that their interactions can be tracked for later analysis. In our work-in-progress second approach we used *SIMVA* as the mediator tool. The teacher can upload IMS CP packages to *SIMVA* and give access to their students form the LMS using LTI links; while seamlessly handling the xAPI statements generated by the packaged games. Although the CMI-5 support is being implemented in *SIMVA* as of this writing, the LTI support lets teachers and researchers that wish to validate their games or conduct experiments easily integrate *SIMVA* in their LMS courses benefitting from functionalities such as group control, pre-post testing, students' progress real-time monitoring, safe trace storage and machine learning analysis on the resulting data.

We are currently working on adding a feature to *uAdventure* that would allow developers to automatically package and games to *SIMVA*, generating an LTI link so that teachers can include the resulting game in their courses. This will require full CMI-5 support from both *uAdventure* and *SIMVA*, resulting in both interoperable games and simple to integrate activities.

ACKNOWLEDGMENT

This work has been partially funded by the Regional Government of Madrid (eMadrid S2018/TCS-4307, co-funded by the European Structural Funds FSE and FEDER), by the Ministry of Education (PID2020-119620RB-I00) and by the Telefónica-Complutense Chair on Digital Education and Serious Games.

REFERENCES

[1] T. M. Connolly, E. A. Boyle, E. MacArthur, T. Hainey, and J. M. Boyle, "A systematic literature review of empirical evidence on computer games and serious games," *Comput. Educ.*, 2012, doi: 10.1016/j.compedu.2012.03.004.

[2] A. Calvo-Morata, C. Alonso-Fernández, M. Freire, I. Martínez-Ortiz, and B. Fernández-Manjón, "Serious games to prevent and detect bullying and cyberbullying: A systematic serious games and literature review," *Comput. Educ.*, vol. 157, p. 103958, 2020, doi: https://doi.org/10.1016/j.compedu.2020.103958.

[3] P. Moreno-Ger, J. L. Sierra, and B. Fernández-Manjón, "Game-Based Learning in e-Learning Environments," *UPGRADE-Novática*, vol. IX, no. 3, pp. 15–20, 2008.

[4] B. Horn, A. K. Hoover, J. Barnes, Y. Folajimi, G. Smith, and C. Harteveld, "Opening the Black Box of Play," in *Proceedings of the 2016 Annual Symposium on Computer-Human Interaction in Play*, 2016, pp. 142–153, doi: 10.1145/2967934.2968109.

[5] E. Barra, A. Gordillo, D. Gallego, and J. Quemada, "Integration of SCORM packages into web games," in *Proceedings - Frontiers in Education Conference, FIE*, 2013, doi: 10.1109/FIE.2013.6684913.

[6] Á. Serrano-Laguna, I. Martínez-Ortiz, J. Haag, D. Regan, A. Johnson, and B. Fernández-Manjón, "Applying standards to systematize learning analytics in serious games," *Comput. Stand. Interfaces*, vol. 50, pp. 116–123, Feb. 2017, doi: 10.1016/j.csi.2016.09.014.

[7] A. Velicanu, I. Lungu, V. Diaconita, and C. Nisioiu, "INTEGRATING SERIOUS GAMES INTO E-LEARNING PLATFORMS: PRESENT AND FUTURE," *9th Int. Sci. Conf. eLearning Softw. Educ. Bucharest, April 25-26, 2013*, no. i, pp. 380–386, 2013.

[8] J. Torrente, P. Moreno-Ger, I. Martínez-Ortiz, and B. Fernandez-Manjon, "Integration and deployment of educational games in e-learning environments: The learning object model meets educational gaming," *Educ. Technol. Soc.*, vol. 12, no. 4, pp. 359–371, 2009.

[9] V. M. Perez-Colado, I. J. Pérez-Colado, M. Freire-Morán, I. Martínez-Ortiz, and B. Fernández-Manjón, "UAdventure: Simplifying narrative serious games development," in *Proceedings - IEEE 19th International Conference on Advanced Learning Technologies, ICALT 2019*, 2019, doi: 10.1109/ICALT.2019.00030.

[10] P. R. Polsani, "Use and abuse of reusable learning objects," *Journal of Digital Information*. 2003.

[11] Á. Del Blanco, E. J. Marchiori, J. Torrente, I. Martínez-Ortiz, and B. Fernández-Manjón, "Using e-learning standards in educational video games," *Comput. Stand. Interfaces*, vol. 36, no. 1, pp. 178–187, 2013.

[12] Learning Technology Standards Committee of the

IEEE, “1484.12.1-2002 - IEEE Standard for Learning Object Metadata,” *IEEE*. 2002.
[13] N. Palavitsinis, N. Manouselis, and S. Sanchez-Alonso, “Metadata quality in learning object repositories: A case study,” *Electron. Libr.*, 2014, doi: 10.1108/EL-12-2011-0175.
[14] F. A. M. Fernández-Diego, M. L. Gordo, A. Boza, L. Cuenca, L. Ruiz, M. M. E. Alemany, “METADATA, REPOSITORY AND METHODOLOGY IN LEARNING OBJECTS,” in *7th International Conference on Education and New Learning Technologies*, 2015, pp. 4755–4761.
[15] M. Tavakoli, M. Elias, G. Kismihók, and S. Auer, “Metadata Analysis of Open Educational Resources,” in *LAK21: 11th International Learning Analytics and Knowledge Conference*, 2021, pp. 626–631, doi: 10.1145/3448139.3448208.
[16] IMS Global Learning Consortium, “IMS Content Packaging Information Model. Version 1.2. Public Draft v2.0,” 2007.
[17] ISO/IEC, “ISO/IEC FDIS 19788-1 Information technology — Learning, education and training — Metadata for learning resources — Part 1: Framework,” 2011.
[18] A. Bakhouyi, R. Dehbi, M. T. Lti, and O. Hajoui, “Evolution of standardization and interoperability on E-learning systems: An overview,” *2017 16th Int. Conf. Inf. Technol. Based High. Educ. Training, ITHET 2017*, 2017, doi: 10.1109/ITHET.2017.8067789.
[19] L. Lindert and B. Su, “The evolution of SCORM to Tin Can API: Implications for instructional design,” *Educ. Technol.*, vol. 56, no. 4, pp. 44–46, 2016.
[20] P. Papazoglou Papazoglakis, “The Past, Present and Future of SCORM,” *Econ. Informatics*, vol. 13, no. 1, pp. 16–26, 2013.
[21] P. C. Víctor Manuel, P. C. Iván José, F. M. Manuel, M. O. Iván, and F. M. Baltasar, “Simplifying the creation of adventure serious games with educational-oriented features,” *Educ. Technol. Soc.*, vol. 22, no. 3, pp. 32–46, 2019.
[22] I. J. Pérez-Colado, V. M. Pérez-Colado, I. Martínez-Ortiz, M. Freire, and B. Fernández-Manjón, “A Scalable Architecture for One-Stop Evaluation of Serious Games,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 12517 LNCS, pp. 69–78, 2020, doi: 10.1007/978-3-030-63464-3_7.