Enhancing Learning Environments by Integrating External Applications

Carlos Alario-Hoyos, Miguel L. Bote-Lorenzo, Eduardo Gómez-Sánchez,
Juan I. Asensio-Pérez, Guillermo Vega-Gorgojo and Adolfo Ruiz-Calleja

Abstract—**This paper discusses the lightweight integration of external applications in different learning environments like LMSs, PLEs or MOOCs using the GLUE! architecture. Also, the current status of GLUE! is presented, describing the particularities of integrating external applications in Moodle, LAMS and MediaWiki. Finally, the paper gives instructions to those interested in trying GLUE! or contributing to the integration of new applications or environments.**

***Index Terms*— Learning Environments, External Applications, GLUE!, LMSs.**

I. Introduction

The centralization of contents, learning activities and assessment activities in one single learning environment became widespread during the last decade with the adoption of LMSs (Learning Managements Systems) like Moodle, LAMS, Blackboard or Sakai in most educational institutions [1]. These environments bring learners together when facing remote activities, facilitating their communication and also their work in collaboration by means of built-in tools such as chats, forums or discussion boards. However, the limited set of built-in tools that are offered by LMSs hinders the enactment of learning situations in which learners are intended to carry out a wide range of activities (e.g. drawing activities, simulations, etc.) [2]. Also, instructors and learners could prefer to employ the external applications they are used to, instead of using those built-in with an equivalent functionality (e.g. using the Facebook chat for the communication with their partners rather than the Moodle chat).

In this context, the GLUE! (Group Learning Uniform Environment) architecture [3] has been proposed to enable a lightweight integration of many external applications like Google Docs (now Google Drive) or Doodle in different learning environments. GLUE! (Figure 1) is made of a central software component called *GLUE! core* and two kinds of *adapters* [4]. The GLUE! core supports most of the integration functionality, including the creation, configuration, retrieval, update and deletion of application instances. The adapters wrap either learning environments or external applications connecting them with the GLUE! core. The selection of the external applications and the management of their instances are done within the learning environment GUI.

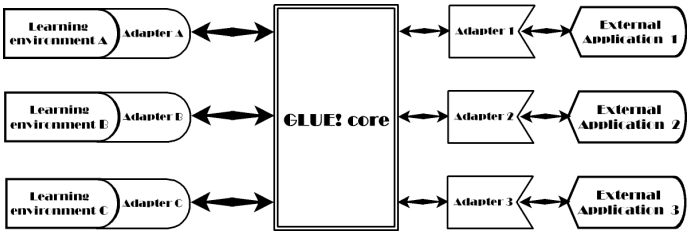

Fig. 1. **Overview of the GLUE! architecture,** including the GLUE! core and the adapters for learning environment and external applications.

II. Integration of External Applications in Different Platforms

In the case of institutional learning environments like LMSs, instructors can use GLUE! to automatically create and configure different instances of each external application for each learner participating in a given activity. Besides, since most LMSs support the arrangement of participants in groups, then every group could receive a different instance to work in collaboration. Afterwards, learners logged in the LMS would find all the applications they need to use (both built-in and external) particularized to carry out their learning activities, individually or in groups.

Ongoing trends on learning environments are fostering more learner-centered software alternatives grouped under the term PLEs (Personal Learning Environments) [5]. PLEs are not expected to replace LMSs, since both models may coexist, being PLEs dominant on informal learning, while LMSs could be preferred by institutions for formal learning [6]. PLEs can also be employed to centralize contents and applications in one single environment, but in this case according to learners' choice. Thus, learners could also employ GLUE! to integrate external applications within the PLE user interface. That integration would be possible due to the multi-tier architectural design of GLUE!, and would only require the development of an adapter wrapping each particular PLE. In this case, learners should be able to decide which applications to integrate in which learning activity. Besides, and if the PLE allowed learners to define group configurations, the own learners could create instances of external applications to be shared by each group with the GLUE! mediation.

This work has been partially funded by the Spanish Ministry of Economy and Competitiveness Projects TIN2008-03-23 and TIN2011-28308-C03-02/03, the Autonomous Government of Castille and León Project VA293A11-2, the Regional Government of Madrid project S2009/TIC-1650, and the postdoctoral fellowship Alianza 4 Universidades.

Carlos Alario-Hoyos is with the Gradient research group (see http://gradient.it.uc3m.es), Universidad Carlos III de Madrid, Department of Telematics Engineering, Av. Universidad, 30, Torres Quevedo Building, Leganes 28911, Spain. Corresponding author e-mail: calario@ it.uc3m.es.

The remaining authors are with the GSIC-EMIC research group (see http://gsic.uva.es) School of Telecommunications Engineering, University of Valladolid 47011, Spain.

An intermediate approach between the institutionally-oriented LMSs and the learner-centered PLEs are the large-scale online courses also referred to as MOOCs (Massive Open Online Courses) [7]. In MOOCs instructors partially determine the learning and assessment activities, as well as the basic content of the course, while learners may add new content, creating connections with other learners. Users of open source environments supporting MOOCs like Class2Go or OpenMOOC might also benefit from the lightweight integration of external applications proposed by GLUE!, highly reducing the time to create and configure different external application instances for such a large number of participants. As an example, instructors may first create generic instances for all the participants in a MOOC, similarly to the case of LMSs. Then, learners might generate particular instances to work collaboratively in large or small groups, as an analogy to PLEs.

III. Current Status of GLUE!

Different adapters have already been developed to integrate external applications in the Moodle, LAMS and MediaWiki learning environments. It is important to note that the functionality offered by the GLUE! core to instructors and learners in order to select external applications and manage external application instances is equivalent no matter the learning environment employed. Nevertheless, the adapters connecting Moodle, LAMS and MediaWiki with the GLUE! core were designed according to the particular features offered by each of these environments, pursuing a more seamless integration as compared to built-in tools.

Taking Moodle as an example, the adapter adds a special *Moodle activity* that allows instructors to choose any available external application from a drop-down menu. Instructors can request the creation, configuration, update, retrieval and deletion of external application instances within the Moodle graphical interface. It is particularly relevant to say that these instances are automatically assigned to each Moodle group or grouping, providing instructors with high efficiency when managing a course that employs external applications (e.g. just by pressing a button within the Moodle interface instructors can create a different Google Documents instance for each of the groups defined in that learning activity). Also, instructors can reuse instances of the same application in different activities of the same Moodle course, so that, for example, a group may review their partners' work output of a former activity. Learners can access the instances depending on their group or grouping settings, in order to perform individual or collaborative activities (see Figure 2 top). Meanwhile instructors can monitor learners' performance at any moment by visualizing the available instances, and giving feedback as the learning activities are being carried out.

A similar process occurs in LAMS, where instructors can add external applications within the LAMS authoring environment. The LAMS adapter adds a new *LAMS tool* that allows instructors to select any of the available external applications, including them in the sequence of activities defined for a LAMS lesson, as usual. This sequence may comprise any of the LAMS built-in tools plus any available external application. Also, instructors using GLUE! can benefit from the LAMS features for creating groups, branches, conditions or stop gates no matter if they are adding built-in tools or integrating external applications. Afterwards, instructors deploy their lessons in the LAMS monitoring environment, being instances of external applications automatically created as defined in the LAMS authoring environment (see Figure 2 bottom). Here, instructors can also monitor learners' work by accessing external instances as in the case of Moodle. Interestingly, those LAMS lessons that include integrated applications can be exported and shared among instructors as long as they import them in the same LAMS installation or in another one with the same available external applications.

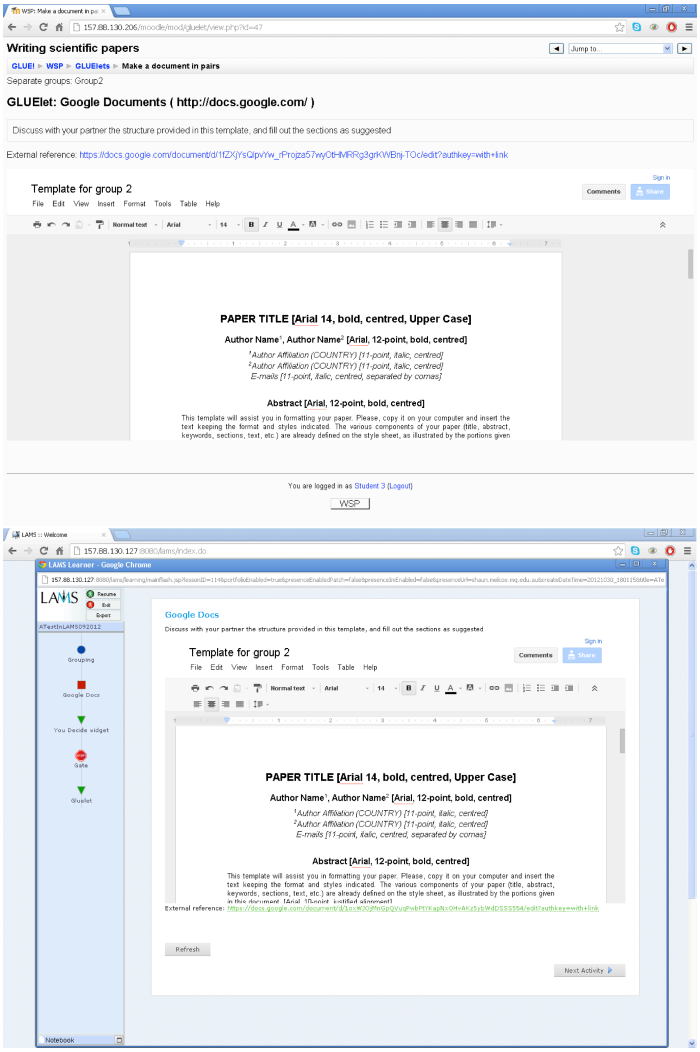

Fig. 2. Screenshots showing the integration of **Google Documents** in **Moodle** (top) and **LAMS** (bottom)

The case of the MediaWiki Content Management System is somewhat different, since it relies on a non-hierarchical generation of content and resources. The novelty in this environment is that any user (no matter if playing the role of

instructor or learner) could request the creation and configuration of external application instances with the mediation of the GLUE! architecture. Thus, each learner would be responsible for the content published and the applications selected in order to carry out the learning activities, as an analogy to the case of PLEs.

Authentic experiments with real instructors and learners were conducted employing Moodle, LAMS and MediaWiki as the centralized learning environments, and GLUE! as the integration approach [3]. These experiments showed a significant reduction of the time demanded to instructors when deploying learning situations with a non-trivial group structure that require the integration of external applications. For example, instructors took 7.5 minutes to deploy in Moodle a complex course with five collaborative activities, three external applications, 36 different groups and groupings and 72 instances, thanks to GLUE! capabilities [3]. The same deployment is possible creating the instances through the application graphical interface and then copying and pasting URLs in Moodle, but that takes 42.5 minutes, almost five times more. Besides, these experiments evidenced that the integrated applications satisfied learners' needs, who particularly highlighted the benefits of having all the required applications in one single learning environment.

At this time, GLUE! has already been demonstrated in LMS-specific LAMS and Moodle conferences, with great interest from their respective creators. Anyone can try GLUE! using either of these LMSs and also MediaWiki, requesting a trial account in the GLUE! website (see http://gsic.uva.es/glue). Moreover, the binary code of the GLUE! core and several examples of adapters can be downloaded in a full package or in individual packages, and installed in order to run GLUE! within Moodle 1.9x or 2.x, LAMS 2.x and MediaWiki 1.x versions.

Finally, it is noteworthy that GLUE! is an open source project. The source code of GLUE! can be employed as the basis for those that want to develop new adapters or improve the existing ones. Interestingly, the development of new adapters for external applications that offer a web API is estimated in about one hundred new source lines of code and six to eight programming hours (if reusing the existing code) [3]. That is far less than what is needed to implement an ad hoc integration in for instance Moodle or LAMS.

IV. Conclusion

This paper has discussed how the GLUE! architecture can be useful to integrate third-party external applications in different learning environments. Depending on the particular environment this integration could be made by the instructor or by the learners. For instance, while in institutionally-oriented LMSs like Moodle, the instructor should select the suitable tools to be employed in the learning activities; in PLEs, learners should be responsible for these settings. In any case, the GLUE! architecture positions as an open source, lightweight and low effort alternative to integrate external tools, especially in scenarios where collaboration and group work are required.

Acknowledgments

Authors would like to thank David A. Velasco-Villanueva, Javier Aragón and Javier Hoyos for their work on the development of the GLUE! core and some of the adapters.

References

[1] Coates, H., James, R. and Baldwin, G., A critical examination of the effects of learning management systems on university teaching and learning, Tertiary Education and Management, 11(1):19-36, 2005.

[2] Bower, M. and Wittmann, M., A Comparison of LAMS and Moodle as Learning Design Technologies - Teacher Education Students' Perspective, Teaching English with Technology, Special Issue on LAMS and Learning Design, 11(1):62-80, 2011.

[3] Alario-Hoyos, C., Bote-Lorenzo, M.L., Gómez-Sánchez, E., Asensio-Pérez, J.I., Vega-Gorgojo, G. and Ruiz-Calleja, A., GLUE!: An Architecture for the Integration of External Tools in Virtual Learning Environments, Computers & Education, 60(1):122-137, 2013.

[4] Gamma, E., Helm, R., Johnson, R. and Vlissides, J., Design Patterns: Elements of Reusable Object-oriented Software. Addison-Wesley, Reading, MA, USA, 1995.

[5] van Harmelen. M., Personal Learning Environments, In Proceedings of the 6th IEEE International Conference on Advanced Learning Technologies, ICALT 2006, pages 815-816, Kerkrade, The Netherlands, July 2006, IEEE Computer Society.

[6] Wilson, S., Liber, O., Johnson, M., Beauvour, P., Sharples, P. and Milligan, C., Personal Learning Environments: Challenging the dominant design of educational systems, Journal of e-Learning and Knowledge Society, 3(2): 27-38, 2007.

[7] Martin, F.G., Will massive open online courses change how we teach? Communications of the ACM, 55(8):26–28, 2012.

Carlos Alario-Hoyos received his MSc and PhD degrees in telecommunications engineering from the University of Valladolid Spain in 2007 and 2012 respectively. He is currently working as a postdoctoral researcher and teaching assistant with the fellowship Alianza 4 Universidades at the Gradient research group in the Department of Telematics Engineering, Universidad Carlos III de Madrid, Leganés, Spain. His main research interests include software applications, architectures and environments for the support of collaborative learning.

Miguel L. Bote-Lorenzo received his MSc and PhD degrees in telecommunications engineering from the University of Valladolid, Spain, in 2001 and 2005 respectively. He is currently an associate professor at the Department of Signal Theory, Communications and Telematics Engineering, University of Valladolid, Spain. His main research interests include distributed systems and their application for the support of learning, with special focus in Computer-Supported Collaborative Learning and tool integration in Learning Management Systems.

Eduardo Gómez-Sánchez received his MSc and PhD degrees in telecommunications engineering from the University of Valladolid, Spain, in 1996 and 2001 respectively. He is currently an associate professor at the Department of Signal Theory, Communications and Telematics Engineering, University of Valladolid. His main research interests include distributed systems and their application for collaborative learning.

Juan I. Asensio-Pérez received his MSc and PhD degrees in telecommunications engineering from the University of Valladolid, Spain, in 1995 and 2000 respectively. He is currently an associate professor at the Department of Signal Theory, Communications and Telematics Engineering, University of Valladolid. His main research interests include technology-enhanced learning, with special focus on the development of tools for the support of design processes for the creation of Computer-Supported Collaborative Learning scripts, and distributed systems architectures for the integration of service-like tools in Learning Management Systems.

Guillermo Vega-Gorgojo received his MSc and PhD degrees in telecommunications engineering from the University of Valladolid, Spain, in 2000 and 2007 respectively. He is currently an associate professor at the Department of Signal Theory, Communications and Telematics Engineering, University of Valladolid. His main research interests include Semantic Web technologies and distributed systems for the support of learning.

Adolfo Ruiz-Calleja received his MSc degree in telecommunications engineering from the University of Valladolid, Spain, in 2007. He is currently a PhD student and a teaching assistant at this university. His main research interests include Linked-Data, semantics and their application to learning scenarios.