OpenDSA/Mastery Grids Exercise Interchange

Hamza Manzoor
Virginia Tech
mhamza1@vt.edu

Kamil Akhuseyinoglu
University of Pittsburgh
kaa108@pitt.edu

Cliff Shaffer
Virginia Tech
shaffer@vt.edu

Peter Brusilovsky
University of Pittsburgh
peterb@pitt.edu

ABSTRACT

In this paper, we exhibit a successful integration of multiple smart learning content through Learning Tools Interoperability (LTI) protocol. This work allows OpenDSA and Mastery Grids to serve new content types by leveraging the standardized protocols and smart learning content servers.

KEYWORDS

interoperability, LTI, learning management system, smart learning content

ACM Reference Format:
Hamza Manzoor, Kamil Akhuseyinoglu, Cliff Shaffer, and Peter Brusilovsky. 2019. OpenDSA/Mastery Grids Exercise Interchange. In *Proceedings of SPLICE'19*. ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

OpenDSA [5] is an e-Textbook project developed at Virginia Tech, which provides materials to support courses in a wide variety of Computer Science-related topics. Mastery Grids [4] is open-source progress visualization environment which has open (social) learner model features designed at the University of Pittsburgh.

In this work, we follow the vision of the ACM ITiCSE working group [2] on integrating smart learning content using standardized protocols, namely Learning Tools Interoperability (LTI) protocol. We demonstrate the integration and reuse of smart learning contents which are developed and served by multiple universities. We present steps towards integrating smart content served through ACOS server [6], which is a smart learning content server, into OpenDSA. Similarly, we extended contents served by Mastery Grids by integrating CodeWorkout [3] small programming problems and visualize student progress in an open learner model (OLM).

2 BACKGROUND

OpenDSA provides materials to support courses in a wide variety of Computer Science-related topics and allows users to choose from available modules to put together a custom textbook for a course and that book can be exported to Canvas as individual modules and assignments. OpenDSA contains Java exercises and most of these exercises are developed by researchers and students at Virginia Tech. Similarly, many other researchers have worked on the similar projects and have their own versions of exercises. For example, ACOS server, which is a smart learning content server developed as a joint project of Aalto University and the University of Pittsburgh, enhances the re-usability of online learning activities by decoupling the content and the existing interoperability protocols including LTI. ACOS is capable of serving multiple smart contents including Java and Python animations and exercises. OpenDSA does not have Python exercises and one of the solutions was to develop these exercises at Virginia Tech, which will require months of work or we could somehow integrate the already built exercises in OpenDSA such that they seem like a part of OpenDSA.

Mastery Grids is open social learner modeling interface which shows students's progress or knowledge in particular topics and different content types, let students to compare themselves with other learners and provide navigation support. CodeWorkout is an open source system which helps students to practice Java programming problems and it is an LTI-provider. Mastery Grids has been used as a practice environment in many classroom experiments including introductory Java programming courses. However, it needs to serve Java coding exercises for advanced programming topics to be used in advanced programming courses. Without integrating already existing tools and content, we may need to develop new type of learning tool and author learning content.

3 INTEGRATION DETAILS

3.1 Integration of ACOS Content into OpenDSA

In OpenDSA, we integrated the Python exercises (Parson's problems) and Java and Python animations. This integration has been possible because OpenDSA works as both, the tool consumer and the tool provider. It acts as a tool provider to Canvas to serve the textbooks and within those books, OpenDSA calls other tools such as CodeWorkout to serve different types of exercises. We added ACOS as a new tool within OpenDSA to serve the Python exercises and animations of Java and Python.

3.1.1 Integrating Python Exercises.

The Python Parson's problem are added in OpenDSA and since these are exercises, the scores received on these are reported back

SPLICE'19, February 27th, 2019, Minneapolis, MN, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

Figure 1: Parson's problem served through Canvas via OpenDSA eTextbook.

to the LMS through LTI. Figure 1 shows an example of Parson's problem served through Canvas via OpenDSA eTextbook.

3.1.2 Integrating Python and Java Animations into OpenDSA.

ACOS also serves various animations to understand Java and Python concepts. These animations show each process step by step such as, the modification in stack when a certain line of code is executed and how the output is calculated. Figure 2 shows an example animation of String concatenation in Java.

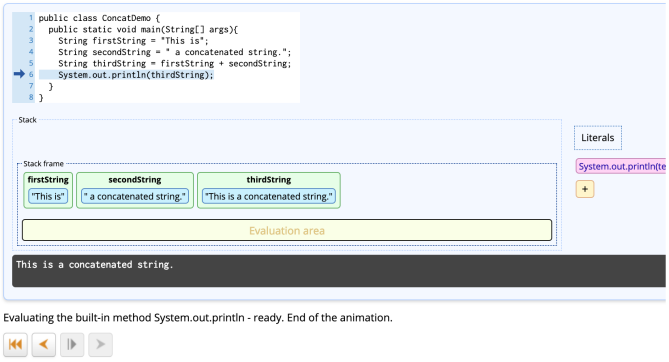

Figure 2: Animation of String concatenation in Java.

3.2 Integrating CodeWorkout Exercises into MasteryGrids

Mastery Grids is capable of accessing multiple types of smart learning content through a proprietary protocol [1]. However, in this work, Mastery Grids is enhanced by integrating and accessing learning content through LTI protocol. This enhancement puts Mastery Grids into highest level of smart content delivery platform as declared in [2] and makes it an LTI-consumer. However, we need to include more learning content to cover advanced topics. Authoring smart learning content to fulfill the requirements of programming courses at different levels (e.g. beginner, advanced) requires plenty

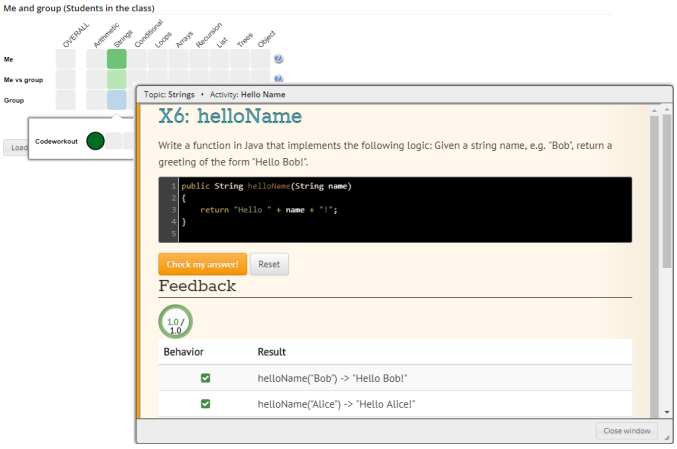

Figure 3: An instance of a CodeWorkout exercise accessed from Mastery Grids interface.

of time. Fortunately, integrating already existing tools and learning content through standardized protocols such as LTI is the solution to this problem. To implement this solution, we have selected CodeWorkout system to demonstrate the importance and applicability of smart content integration. CodeWorkout is an LTI-provider and has many Java programming exercises in advanced topics which makes it one of the first possible learning content to be integrated easily and effortlessly.

We have created a practice environment for Java learners with multiple topics such as Recursion, Trees etc. (see Figure 3). A student who accessed the Mastery Grids after a logged in, can access CodeWorkout exercises without being required to log in to CodeWorkout system. After the student completed and submitted her answer successfully, the student modelling service which is already integrated to the Mastery Grids is updated automatically and the student can view her progress updated (colored squares). This shows that CodeWorkout exercise integration through LTI protocol does not eliminate the already existing features of the Mastery Grids.

REFERENCES

[1] P. Brusilovsky. A distributed architecture for adaptive elearning. In *Proc. of WWW2004, the Thirteen International World Wide Web Conference, 2004*, 2004.
[2] P. Brusilovsky, S. Edwards, A. Kumar, L. Malmi, L. Benotti, D. Buck, P. Ihantola, R. Prince, T. Sirkiä, S. Sosnovsky, et al. Increasing adoption of smart learning content for computer science education. In *Proceedings of the Working Group Reports of the 2014 on Innovation & Technology in Computer Science Education Conference*, pages 31–57. ACM, 2014.
[3] S. H. Edwards and K. P. Murali. Codeworkout: short programming exercises with built-in data collection. In *Proceedings of the 2017 ACM Conference on Innovation and Technology in Computer Science Education*, pages 188–193. ACM, 2017.
[4] T. D. Loboda, J. Guerra, R. Hosseini, and P. Brusilovsky. Mastery grids: An open source social educational progress visualization. In *European conference on technology enhanced learning*, pages 235–248. Springer, 2014.
[5] C. A. Shaffer, V. Karavirta, A. Korhonen, and T. L. Naps. Opendsa: beginning a community active-ebook project. In *Proceedings of the 11th Koli Calling International Conference on Computing Education Research*, pages 112–117. ACM, 2011.
[6] T. Sirkiä and L. Haaranen. Improving online learning activity interoperability with acos server. *Software: Practice and Experience*, 47(11):1657–1676, 2017.