

Composing Learning Environments with e-Textbook System

Yuta Taniguchi^[0000-0003-3298-8124], Tsubasa Minematsu, and
Atsushi Shimada^[0000-0002-3635-9336]

Kyushu University, Fukuoka, Japan
taniguchi.yuta.941@m.kyushu-u.ac.jp,
{minematsu,atsushi}@ait.kyushu-u.ac.jp

Abstract. The introduction of ICT technology into education has been attracting attention, and various educational platforms and tools have been developed. With the emergence of standards such as Learning Tools Interoperability, it has become possible to call external tools from Learning Management Systems in a unified manner. However, it is still challenging to integrate the tools and use them as a seamless learning environment. For this reason, people tend to develop a monolithic tool that integrates functions that meet individual needs. Repeatedly reimplementing similar functions leads to the mess of various learning log formats and data dispersion across the tools, making it hard to realize learning analytics across multiple learning support systems of different educational institutions. In order to solve this problem, we propose the concept of Compositional Learning Environments, with which we can combine any tools to form a new and more complex one. This demonstration shows two practical examples of combinations of an e-textbook tool with other types of learning support tools.

Keywords: Learning environment · Design framework · Learning Tools Interoperability.

1 Introduction

Nowadays, learning environments are composed of several online systems, including Learning Management Systems (LMSs) providing fundamental functions for any classes, electronic textbook systems [2] widely used in lecture-style classes, and programming environments used in programming exercise classes. In addition, in the past decade, Learning Analytics (LA) has been attracting attention. LA has revealed the value of the educational data. Researchers have developed a variety of learning support systems that can record activities and outcomes of teachers and learners, as well as systems that provide feedback based on analysis of those data.

Copyright © 2021 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

However, it is difficult for a single learning environment to cover the needs of multiple class formats, such as face-to-face and online, as well as the different needs of different subjects, teachers, and students. Therefore, there is a need for a learning environment that one can flexibly configure according to their requirements.

Making use of plugins is one way to customize LMSs. However, plugins have the disadvantage of being closely dependent on a particular LMS, which means the format of the recorded educational data also depends on the LMS, which makes cross-platform analysis difficult from the viewpoint of LA. An alternative approach currently available is to use Learning Tools Interoperability (LTI). LTI is the standard developed by the IMS Global Learning Consortium, which defines the conventions for calling and linking LTI-compliant Web applications (LTI tools) as external tools from an LMS. LTI makes it possible to call the same external tool from different LMSs in a unified manner and also makes it possible to easily link relatively complex systems, even electronic textbook systems, with LMSs. Although LTI has made it possible to link LMSs with external tools, as far as the authors know, there is no definitive way to link external tools with each other. Because each LTI tool is an independent Web application, it is still difficult to use multiple external tools as a seamless learning environment.

In order to solve these problems, we propose the concept of *Compositional Learning Environments (CLE)*. CLE is a conceptual framework for combining independent LTI tools to form a more complex LTI tool. With CLE, we aim to promote the simplification and modularization of LTI tools, which would then realize lower development costs and a shared data format of educational data for the same functionality. This demonstration will present real examples of CLE-based compound learning environments in which an e-textbook system is combined with other systems.

2 Compositional Learning Environments

The core function of CLE as a concept is to create one new LTI tool from two or more LTI tools. It realizes the integration of tools visually, functionally, or both. The aim of the CLE is twofold. The first is modularity. In the past, many learning support systems were designed as monolithic systems. CLE allows us to design the learning environment as a combination of reusable modules, and each module can be used as an independent learning tool. The second is to control the diversity of data formats. In order to achieve cross-class and cross-institutional LA, it is important that the format of the data used is uniform. Using the same module (LTI tool) for a particular function, such as reading textbooks, we can ensure that the common data format is used for recording the same learning activity.

We developed an implementation called *CLE Tool* for the proof of concept. CLE Tool can realize visual combinations of LTI tools, which display given tools inside itself in a specified layout. The CLE Tool itself is also an LTI tool, and

The figure shows two overlapping windows. The left window, titled 'BookRoll', displays an 'if文' (if statement) section. It includes the following code snippet:

```
if ( 条件式 ) {
    ブロックA
}
else {
    ブロックB
}
```

Below the code is a flowchart for the 'if文'. A diamond-shaped decision box labeled '条件式' (Condition) has two paths: 'true' leading to 'ブロックA' (Block A) and 'false' leading to 'ブロックB' (Block B). A legend explains: 'true: 真(条件が成立)' (true: condition is true) and 'false: 偽(条件が不成立)' (false: condition is false). Below the flowchart is a list of rules:

- 「条件式」が
 - 真(true)ならばブロックAを,
 - 偽(false)ならばブロックBを実行
- else部分は省略可能

The right window, titled 'WEVL', shows a code editor with the following C code:

```
1 #include <stdio.h>
2
3 int main(void) {
4     if (15 % 3 == 0) {
5         printf("OK");
6     }
7     return 0;
8 }
```

Below the code is a '実行' (Execute) button and a '標準入力' (Standard Input) field. The output shows 'プログラムの結果 成功' (Program result: Success) and 'コンパイラの結果 成功' (Compiler result: Success). At the bottom, there is a chat input field labeled 'チャット (Ctrl+Enterで送信)' and a chat display area.

Fig. 1. Using the e-textbook system BookRoll (left) in combination with a programming learning support system WEVL (right).

thus, by using CLE tools in a nested manner, complex layouts can be realized in a controlled manner.

2.1 Example with Programming Learning Support System

As the first practical example, we present a configuration example for online classes of a programming exercise course. Figure 1 shows the actual screen, where two LTI tools are combined; the left side is the programming learning support environment called WEVL developed by the authors, which is used in programming exercise; the right side is an e-textbook system called BookRoll [1], which is used to display lecture slides.

In addition, the CLE tool provides a real-time chat as one of its features. By posting a message via the text box shown at the bottom of the figure, students can provide feedback to the teacher in real-time. Since the simplicity of a system is vital for online classes to avoid confusion, making it easy to switch between the two is a significant advantage.

2.2 Example with Learning Dashboard System

Here we present another combination of the e-textbook system and learning dashboard system [3]. The learning dashboard system was developed to aggregate and analyze the learning logs collected on BookRoll in real-time. It pro-



Fig. 2. Using the e-textbook system BookRoll (right) in combination with a learning dashboard system (left).

vides valuable feedback to teachers and students for improving learning during lectures. In class, the dashboard system is displayed beside BookRoll, as shown in Fig. 2.

Teachers and students take actions for improvement using the information displayed in the learning dashboard during the lecture. For effective learning improvement, the use of the learning dashboard should not interfere with teaching and learning as much as possible. One of the findings of the previous study [3] is that the simultaneous use of multiple independent tools is a burden to users. This could be due to the need to switch between different tools on one screen, and hence we introduced CLE for the simpler user interface.

3 Conclusion

In this paper, we proposed the concept of Compositional Learning Environments (CLE) and presented two examples of the actual configurations combining an e-textbook system with other types of learning support systems. The CLE concept provides a design philosophy for learning environments in which simple tools are combined to create a more complex tool. By using CLE, the learning environment can be easily, flexibly configured to meet various class needs. Therefore, we believe that simplifying compound learning environments based on the CLE concept would make it easy to use e-textbook systems with different systems.

Acknowledgment

This work was supported by JSPS KAKENHI Grant Number JP21K17863.

References

1. Flanagan, B., Ogata, H.: Integration of learning analytics research and production systems while protecting privacy. In: The 25th International Conference on Computers in Education, Christchurch, New Zealand. pp. 333–338 (2017)
2. Ogata, H., Yin, C., Oi, M., Okubo, F., Shimada, A., Kojima, K., Yamada, M.: E-book-based learning analytics in university education. In: International Conference on Computer in Education (ICCE 2015). pp. 401–406 (2015)
3. Owatari, T., Shimada, A., Minematsu, T., Hori, M., Taniguchi, R.: Real-time feedback dashboard for students in online class. In: 2020 IEEE An International Conference on Engineering, Technology and Education (TALE2020). pp. 953–959 (12 2020)