

構成的学習環境

谷口 雄太^{1,a)} 峰松 翼^{1,b)} 島田 敬士^{1,c)}

概要: ICT 技術の教育への導入が注目され、多様な教育プラットフォームやツールの開発が行われている。LTI などの標準規格の登場により LMS から統一的な方法で外部のツールを呼び出せるようになったが、ツール同士を統合してシームレスな学習環境として利用することは依然として困難である。そのため、個々のニーズにあった機能を統合した単一ツールの開発が行われがちであり、類似機能の開発が繰り返されることで無駄なコストが発生している。また、類似機能の再実装は、データ形式の不統一やデータの分散を招くため、複数の学習支援システムを横断するラーニングアナリティクスの実現を困難にする。本研究ではこの問題を解決するために、Compositional Learning Environments を提案する。本稿では概念実証として、複数の LTI ツールをシームレスに統合する具体的な実装について説明し、実際の授業での利用を念頭においた構成例について報告する。

Compositional Learning Environments

1. はじめに

ICT 技術の教育への導入が注目され、多様な教育プラットフォームや学習支援・分析ツールの開発が行われている。LMS (Learning Management System) はそうしたプラットフォームの 1 つであり、Moodle や Sakai, Canvas などが存在し、大学をはじめとする教育機関で活用されている。LMS はコースや学習者の管理、課題の提出や採点、教材の配信など多くの役割を担っている。

一般に LMS を利用する科目や授業形態は多様であり、学習者や教師は様々な異なるニーズをもっている。必要とされる全ての機能を LMS が網羅的に提供することは困難であり、そのため一般的にはプラグインと呼ばれる機構が LMS に用意されている。プラグインは後からシステムに追加可能なソフトウェアパッケージの一種であり、LMS が持つ機能を強化したり新たな機能を追加することができる。例えば Moodle のプラグインリポジトリ^{*1}には 2020 年 4 月現在で 1,600 を超えるプラグインが登録されている。

プラグインの欠点の 1 つは、特定の LMS に密接に依存している点である。そのため同じプラグインを異なる LMS

で再利用することは基本的にはできず、LMS ごとに同一機能の開発を繰り返す必要がある。別の欠点はインストールが LMS システム全体に影響を与える点である。LMS は通常、教育機関全体で利用されるため、新たなプラグインの導入による問題を最小限に留めるためには、事前の入念なテストと、システムを停止してのインストールといった手順が必要になる。事前に多様なニーズを吸い上げ、必要なプラグインをインストールしておくことは難しく、また新たなプラグインも随時登場するため、LMS のメンテナンスを難しくする。また、これらの欠点はプラグインの開発コストや管理コストを上昇させるばかりでなく、プラグインが記録する学習ログデータの形式が LMS ごとに違ってしまう可能性を増大させてしまう。これは Learning Analytics (LA) の観点からは、教育機関を横断するような分析を阻害するため好ましくない。

プラグインに代わる LMS 拡張方法として Learning Tools Interoperability (LTI)[1] が登場し、導入が進んでいる。LTI は IMS Global Learning Consortium が策定した標準規格で、LTI に準拠した Web アプリケーション (LTI ツール) を LMS から外部ツールとして呼び出し連携するための規約について定めている。図 1 は LTI を用いて同一の外部ツールを複数の LMS から利用している様子を表している。このように LTI 規格の登場により、異なる LMS からでも統一的な方法で外部ツールを呼び出せるようになった。ま

¹ 九州大学 システム情報科学研究院

a) yuta.taniguchi.y.t@gmail.com

b) minematsu@limu.ait.kyushu-u.ac.jp

c) atsushi@ait.kyushu-u.ac.jp

*1 <https://moodle.org/plugins/index.php>

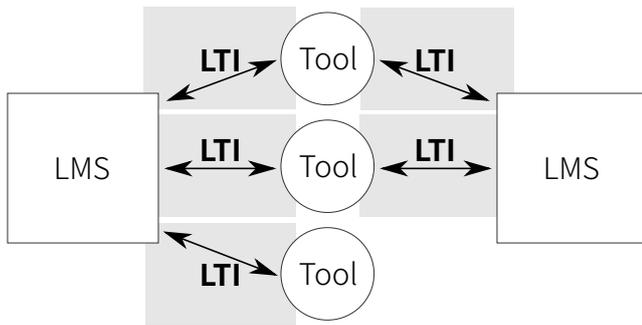


図 1 LTI による LMS と外部ツールとの連携を表している。灰色の部分が LTI がカバーする範囲を表す。ユーザーからは、各外部ツールは独立して動作する個別のツールとして見える。

たプラグインという枠組みに縛られる必要がなくなったことで、例えば電子教科書システム [2] のような、複雑でプラグインの範疇に収められないものであっても容易に LMS と連携させることが可能となっている。

LTI により LMS と外部ツールを連携させることは可能となったが、外部ツール同士の連携については著者らの知る限り依然として決定的な方法は存在していない。一般的にそれぞれの外部ツールは互いに独立した Web アプリケーションに過ぎず、LTI を用いたとしても個別のツールを切り替えながら利用している形となる。そのため複数の外部ツールをシームレスな学習環境として利用することは依然として困難である。複数の機能を併せもつダッシュボードシステム [3] が多く提案されていることを考えれば、まとまりのないツール群を集約して扱いたいというニーズは大きいものと考えられ、現状はこのニーズに答えることができていないと言える。

ダッシュボードのような多機能ツールの開発が個々のニーズに合わせて行われることで、類似機能の開発が繰り返され、無駄なコストが発生する。例えばメモのような機能は、電子教科書やプログラミング学習環境、LMS などをまたいでいつでも利用されうるものであり、汎用性が高い。また、オンライン授業のような状況においては、教師と学習者間でコミュニケーションをとるためのチャットやフォーラムのような機能が必要になるが、こういった機能も同様に汎用性の高い機能の 1 つである。こうした機能を特定のツールの一機能として実装してしまうとデータ記録方式が統一されず、複数の学習支援システムを横断する LA の実現を困難にしてしまう。加えてオンライン授業においては、学習者は限られた画面スペースの中に動画配信や LMS、外部ツールなどを表示せねばならず、これらを頻りに切り替えながら使うことは心理的負担も大きくなる。

本研究ではこれらの問題を解決するために、**構成的学習環境 (Compositional Learning Environments; CLE)** を提案する。CLE は独立した単機能の LTI ツール同士を組み合わせることで、より複雑な LTI ツールを構成するための概念的枠組みである。組み合わせを可能とすることで、外

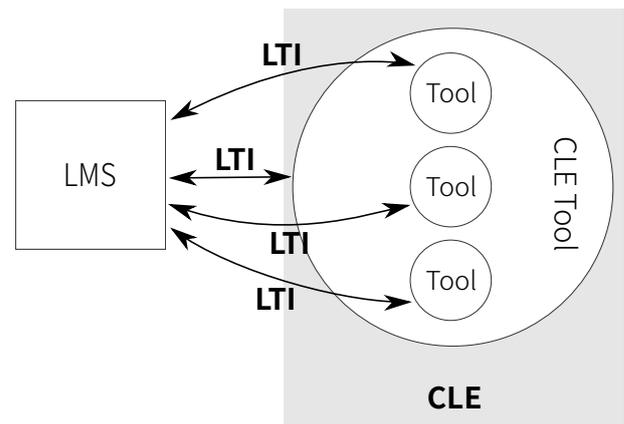


図 2 CLE による複数の外部ツール群の連携を表している。灰色の部分が CLE がカバーする範囲を表す。個々の外部ツールは依然として LMS と繋っているものの、CLE ツールにより一体化され、ユーザーからは単一の外部ツールとして見える。

部ツールの単機能化を促し、再利用性を高めることができる。LMS を拡張する立場から提案された LTI とは異なり、CLE は LTI ツールの開発と利用を改善する立場からの提案といえる。本稿では、CLE 概念の説明を行い、概念実証のための具体的な CLE 実装例について報告する。複数の LTI ツールを単一の LTI ツールに統合する具体的な実装について説明し、実際の授業での利用を念頭においた統合例について説明する。

2. CLE 概念

外部ツールを、LA のためのデータ取得センサーとして考えた場合、自然な形でできるだけ多くのデータを取得できることが望ましい。そのためにはツールが多く利用されることが重要であり、一定の利便性を担保することが重要となる。CLE 概念は**組み合わせ可能性**と**閉性**という 2 つの性質からなり、これらによって教師と学習者の双方に利便性を提供する。

まず組み合わせ可能性とは、2 つ以上の LTI ツールを組み合わせることができるという性質である。LTI ツールは本来独立して動作するように開発されているが、ここでいう組み合わせとは組み合わせられるツールが視覚的におよび/または機能的に統合されていることを指す。

組み合わせには 2 つの種類を考えることができ、1 つは**並列的組み合わせ**、もう 1 つは**入れ子的組み合わせ**である。並列的組み合わせでは、LTI ツール同士は互いに同等の関係にあり、視覚的には例えば図 3 のような平面的な配置やタブを使った配置を考えることができる。一方で入れ子的組み合わせでは、一方のツールは他方のツールといわば親子の関係にある。図 3 の例で言えば、上図のように親が子を埋め込むような配置や、下図のように一部透過させることでオーバーレイする配置を考えることができる。

このように複数のツールを 1 つの学習環境として視覚的

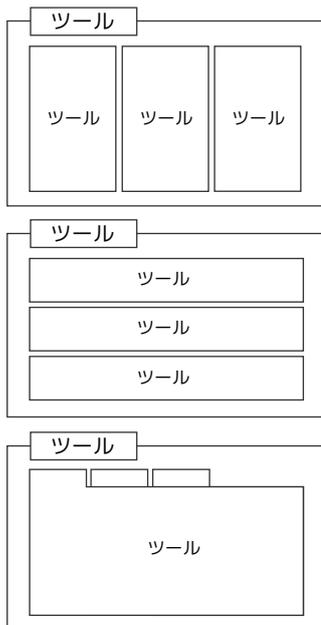


図 3 並列的組み合わせにおける視覚的統合の例.

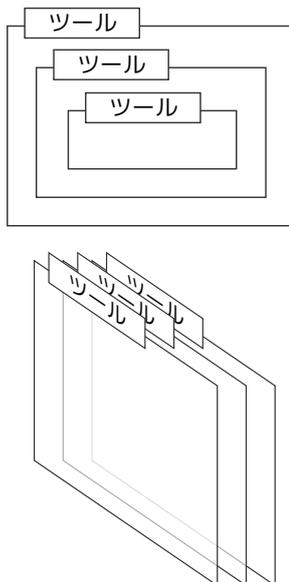


図 4 入れ子的組み合わせにおける視覚的統合の例.

に統合する利点の1つは、ひとまとまりの外部ツール群の利用を学習者に促し、彼らが各ツールを一連の学習活動の中でどのように利用しているかというデータをとれる点である。例えばタブ型のユーザーインターフェイスを用いて、複数のツールを同時に表示できないように制限すれば、ツールの利用時にタブを明示的に切り替えるように学習者に強制することができる。このような設計を行うことで、タブの切り替えを学習ログとして記録することが可能となり、どのツールの後にどのツールを利用しているかという、状態遷移データを取得することが可能となる。

次に閉性とは LTI ツールと LTI ツールとの組み合わせの結果が再び LTI ツールとなることを示す。これはすなわち LTI ツールが上述の組み合わせについて閉じているこ

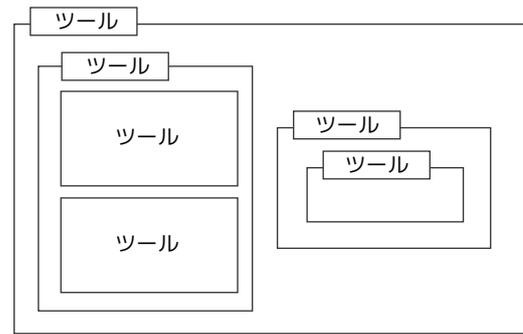


図 5 閉性を利用して多重に組み合わされたツールの例.

とを表す。この性質により例えば図5のように、単純な組み合わせからより複雑な組み合わせを作ることが可能になる。これによりツールの再利用性を格段に高めることができる。

3. CLE 実装

本節では CLE の概念を実証するための実装について述べる。著者らは CLE ツールと呼ばれる外部ツールを実装した。これは、複数の LTI ツールをまとめる「のり付け」の機能と、新たな機能を追加する「機能付与」の機能をもっている。CLE 概念の組み合わせ可能性の言葉でいえば、前者は並列的組み合わせを実現したものに該当し、後者はを入れ子的組み合わせを実現したものとみなすことができる。入れ子的組み合わせは本来は2つの LTI ツールを入れ子状態に組み合わせることだが、ここでは CLE ツールを親、他方のツールを子と考え、子のツールに CLE ツールがもつ機能を付与する形で擬似的に実現する。CLE 概念の閉性については、CLE ツール自体が LTI を実装することにより実現している。

3.1 のり付け

既存の LTI ツールを組み合わせる方法として、CLE ツールは<iframe>を採用した。<iframe>は HTML の要素の1つで、Web ページに別のページを埋め込むことを可能にするものである。組み合わせ対象とする LTI ツールを<iframe>を使って Web ページに複数埋め込むことで、単一の CLE ツール内に統合されたツール群を表示することができる。

上述の実装にはいくつかの注意点があることに注意されたい。まず1つ目は<iframe>に指定する URL である。ここに指定する URL は目的とする LTI ツールを起動するものでなければならない。従って、Moodle などの LTI コンシューマ上で利用可能な URL を指定する必要がある。加えて、ユーザーがこの URL にアクセスする権限をもっていることを保証する必要がある。もし、ユーザーが CLE ツールにアクセスできるが、子 LTI ツールにアクセスする権限をもたない場合、CLE ツールを起動してもうまく当該ツ

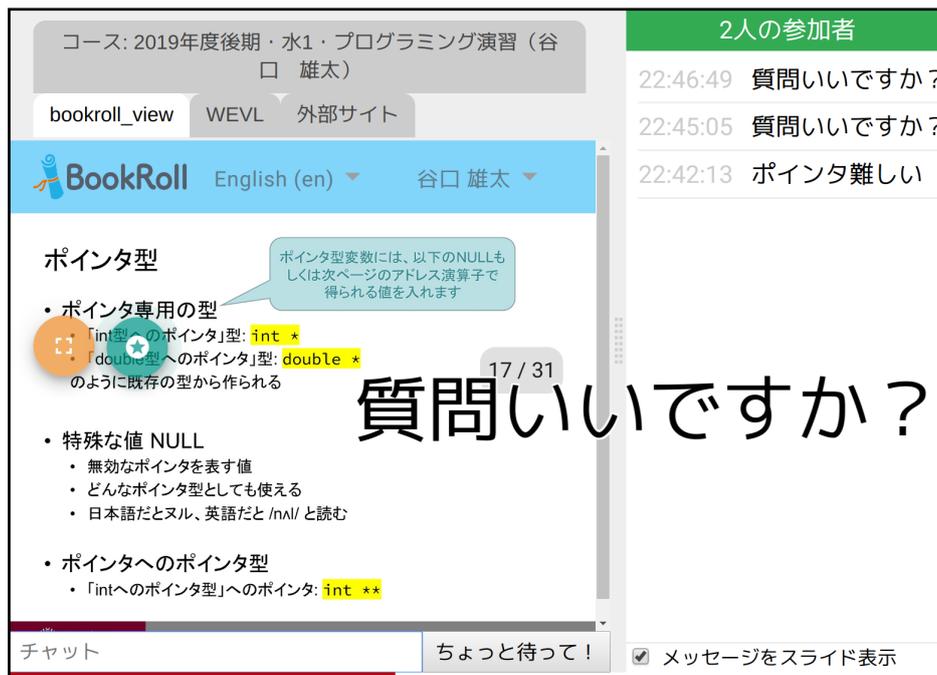


図 6 プログラミング演習科目のオンライン授業向けに構成された実際の CLE ツールの例。3 つの LTI ツールと 1 つの外部 Web サイトが組み合わされており、左上に表示されたタブにより切り替えることができる。また、下のチャット欄からリアルタイムにフィードバックを流すことができる。

ルを表示することはできない。従って、例えば Moodle の場合は、埋め込み対象の LTI ツールと CLE ツールを同じコース内に追加し、追加後に得られる URL を指定することで、これを保証することができる。

2 つ目の注意点は<iframe>の利用に起因するもので、組み合わせるツールによっては<iframe>への埋め込みが許可されていない場合がある。例えば CLE 実装のサイト <https://cle-tool.com/> から、別の LTI ツール <https://other.com/> を<iframe>越しに参照することが <https://other.com/> により禁じられている場合がある。実際、著者らが利用している LTI ツールの 1 つにも同様の制限が掛けられており、そのままでは利用ができなかった。幸い当該ツールは著者らが制御可能であったため、埋め込みを許可するように変更することができた *2 が、より一般的には当該ツールの管理者に<iframe>による埋め込みを許可するよう依頼する必要がある。

このような注意点があるものの、<iframe>を利用する実装には副次的な利点もあり、LTI ツールに限らず任意のサイトを組み合わせることができる。例えばプログラミング演習支援 LTI ツールと、言語リファレンスサイトを統合すれば、演習に必要なリソースを一箇所にまとめることができる。さらに、後述するように Moodle のコースを LTI ツール化すれば、Moodle のプラグインを間接的に組み合わせるようなことも可能である。

*2 リバースプロキシである Nginx の設定で、ツールからのレスポンスに含まれる X-Frame-Options ヘッダを隠すように設定した。

3.2 構成例

実際の例として、ここではプログラミング演習のオンライン授業を念頭においた構成例を紹介する。図 6 は CLE ツールの実際の画面を表示したものである。ここでは、3 つの LTI ツールと 1 つの外部 Web サイトが組み合わされており、左上に表示されたタブにより切り替えることができるようになっている。「コース」から始まるタブは Moodle のコースを LTI ツール化したもので、このタブを表示すれば Moodle がもつ各種プラグインを利用することができるようになっている。選択済みのタブは BookRoll [2] と呼ばれる電子教科書システムであり、このタブを使って講義スライドを表示する。次の「WEVL」と書かれたタブは著者らが開発するプログラミング学習支援環境であり、演習時にはこのタブを開いて演習を進める。最後に「外部サイト」というタブを開くと、C 言語のリファレンスサイトが表示される。

以上の LTI ツールの機能に加え、CLE が付与する機能の 1 つとしてチャット機能がある。下のチャット欄から投稿することで、学生から教師へリアルタイムにフィードバックを流すことができる。このような機能はオンライン授業においては講義内容によらず重要であり、CLE の組み合わせ可能性が生きる場面となる。

4. おわりに

本稿では CLE という概念を提唱し、その概念実証について報告した。CLE 概念は単純なツールの組み合わせによ

り複雑なツールを作るという、学習環境の設計思想を与えるものである。これにより将来開発されるツールをより単機能に特化するよう促すことが本稿のねらいである。本稿で提案した CLE ツールにはいくつかの制約はあるものの、概念実証として十分な機能を備えている。昨今は Moodle のプラグインや LTI ツールをはじめとして、様々なツールが無償で利用できるようになっており、個別に開発する時代から必要なものをピックアップする時代が変わっていく中で、CLE は重要な役割を果たすと信じている。

今後の重要な課題の 1 つは、ツール間の連携方法である。本来独立して設計された個々のツールを必要に応じて、視覚的にだけでなく機能的にも連携させることが必要な場面が考えられる。例えばノートツールについて考えると、教科書の切り抜きやチャットのヒストリ、自身の成績など、各ツールが管理する様々な学習資源をその場で取り込めるようにしたいというニーズがある。このような仕組みを実現するためには、ツール間連携の方法について探る必要があると考えている。

謝辞 本研究は JSPS 科研費 JP17K12804 の助成を受けたものです。

参考文献

- [1] Brown, M., Dehoney, J. and Millichap, N.: The Next Generation Digital Learning Environment, *Verfügbar unter <https://net.educause.edu/ir/library/pdf/eli3035.pdf>* [26.06. 2018] (2015).
- [2] Ogata, H., Taniguchi, Y., Suehiro, D., Shimada, A., Oi, M., Okubo, F., Yamada, M. and Kojima, K.: M2B system: A digital learning platform for traditional classrooms in university, *Practitioner Track Proceedings of the Seventh International Conference on Learning Analytics & Knowledge*, pp. 155–162 (2017).
- [3] Majumdar, R., Akçapınar, A., Akçapınar, G., Ogata, H. and Flanagan, B.: LAViEW: Learning Analytics Dashboard Towards Evidence-based Education, *Companion Proceedings of the 9th International Conference on Learning Analytics and Knowledge*, Society for Learning Analytics Research (SoLAR), pp. 74–79 (2019).