

## LCM (Learner's Controlling Map) / LCM エディタの汎 LMS 化の検討

高橋暁子<sup>†,††</sup> 喜多敏博<sup>†</sup> 中野裕司<sup>†</sup> 合田美子<sup>†</sup>  
鈴木克明<sup>†</sup>

著者らは、eラーニングにおける学習者による学習内容の選択の支援を目的として、学習者向けの学習内容選択支援ツール (LCM) と教授者向けの課題分析図作成支援ツール (LCM エディタ) を開発し、形成的評価によって十分な操作性と有用性を確認した。本研究では、オープンソース LMS である Moodle のモジュールとして開発したこれらのツールについて、他 LMS へ実装するための要件を検討した。その結果、LMS 側の要件として(1)学習コース内でセクション構造を持つこと、(2)セクションごとの評価が可能であること、(3)既存データベースの構造が明確であり、なおかつアクセス権限があること、(4)モジュールによる機能拡張が設計されていることが必要であることがわかった。

### Requirements for LCM (Learner's Controlling Map) / LCM Editor to be cross-LMS

Akiko TAKAHASHI<sup>†,††</sup> Toshihiro KITA<sup>†</sup>  
Hiroshi NAKANO<sup>†</sup> Yoshiko GODA<sup>†</sup>  
and Katsuaki SUZUKI<sup>†</sup>

In the past study, Learner's Controlling Map (LCM) was developed as a learners' tool for selecting learning content using a learning task analysis diagram. LCM Editor was developed as a teachers' tool for using a learning task analysis diagram. As the result of formative evaluation, LCM and LCM Editor have been proved usable without problems. In this study, 4 conditions are presented as the requirements for porting to other LMSs these tools developed as a module of Moodle, one of the common open source LMSs: (1) LMS has section structure in a learning course, (2) the evaluation for every section is possible, (3) the structure of database is clear and the accessing permission is given, (4) the extension of the LMS by modules is possible.

## 1. はじめに

著者らは eラーニングにおける学習者による自己主導的な学習内容の選択の支援を目的に、教授設計理論であるインストラクショナルデザインにおける課題分析図に基づく学習内容選択支援ツール (Learner's Controlling Map: 以下, LCM) [1]を開発した。また、課題分析手法を知らないが、科目内容に関しては専門的知識を有する教授者を対象とした課題分析図作成ツール (LCM エディタ) [2]を開発した。これらのツールは、オープンソース LMS (Learning Management System) として普及している Moodle のモジュールとして実装したが、最終的にあらゆる LMS で動作することを目指している。

そこで本研究では、Moodle のモジュールとして実装した LCM および LCM エディタについて、他 LMS への実装可能性を検討した。

## 2. 先行研究

### 2.1 課題分析図とは

インストラクショナルデザインにおける学習課題分析 (以下、単に「課題分析」と呼ぶ) とは、教材のゴールとして設定した学習目標を修得するために必要な要素とその関係を明らかにする方法である[3]。課題分析の結果を図示したものを課題分析図と呼ぶ。表 1 に示すように、学習課題の種類に応じた分析手法がある。

本研究および関連研究では、「知的技能」の学習課題に着目した。知的技能とは、事物の分別、ルールや原理の適用、問題解決のようにシンボルを使いこなす能力[4]である。言語の使用 (たとえば、作文) のような初歩的な技能から、高度の工学的技能 (たとえば、橋の応力計算) や経済学などの専門的 skill (たとえば、通貨切り下げ効果の予測) まで、幅広い分野に及び、正規の教育のもっとも基礎的な根幹をなしているとされている[4]。よって、既存の eラーニングにおいても知的技能を対象とした多数の学習コースが存在すると考えられる。

知的技能の課題分析手法は「階層分析」である。階層分析の結果である「学習階層図」の例として「いろいろな大きさの整数の引き算 (図 1)」を示す。教授者による階層分析の過程では、最終的に達成すべき学習目標を最上位に置き、「前提条件となる学習目標は何か」を問いながら下位の学習目標を記述していく。前提条件とは、対象の学習目標に先立って学ばれる課題である[4]。したがって学習者が学習する際には、下位から上位へ向かって学習することとなる。分岐箇所においては、どの学習課題か

<sup>†</sup> 熊本大学大学院社会文化科学研究科教授システム学専攻  
Graduate School of Instructional Systems, Kumamoto University

<sup>††</sup> 株式会社エン

See and Inc.

ら学習を開始しても問題はないと言える。

著者らは、自己主導的な学習を支援する立場に立ち、学習者自身に学習内容の選択をゆだね、システム側からは適切な学習内容の選択の支援にとどめたいと考え、課題分析図に基づく学習内容選択支援ツール「LCM」を開発した。

また、学習者に学習内容の選択をゆだねる前提として、適切な知的技能の課題分析図が必要となる。そこで、課題分析手法を知らないが、科目内容に関しては専門的知識を有する教授者用の課題分析図作成ツール「LCM エディタ」を開発した。

表 1 主な学習課題と分析手法 (鈴木 2002[3]の表 5-1 を一部改変)

学習課題	分析手法
言語情報	クラスター分析
知的技能	階層分析
運動技能	手順分析
態度	階層/手順分析/クラスター分析

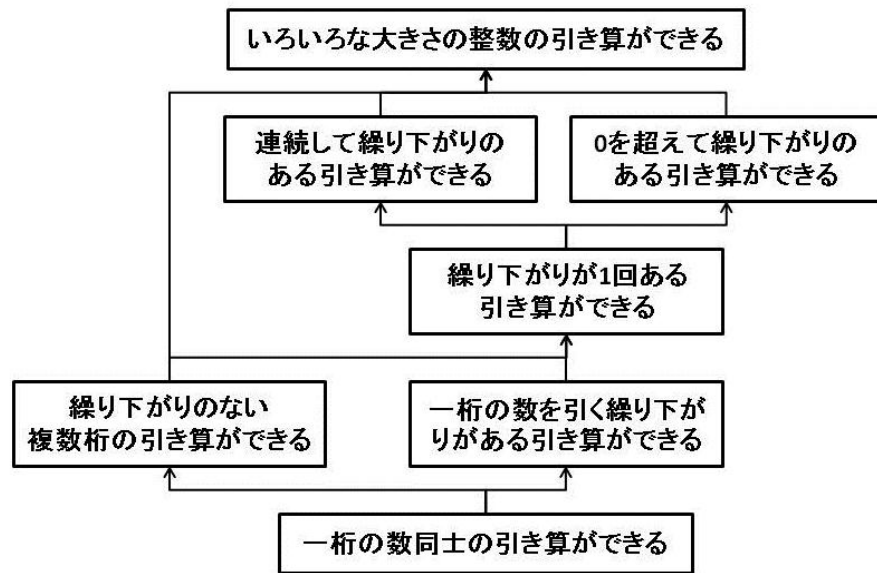


図 1 学習階層図の例「引き算」(鈴木 2002[3]の pp.65 を一部改変)

## 2.2 利用の流れ

LCM および LCM エディタは、Moodle の Block モジュールとして開発した。Moodle のシステム管理者が通常の Block モジュールと同様の方法でインストールすることで、利用可能となる。LCM および LCM エディタの利用の流れを図 2 示す。破線は利用者の操作の流れを意味し、実線はシステム内のデータの流れを意味する。

教授者は Moodle の学習コースに編集モードでアクセスし、LCM エディタを起動する。そして LCM エディタを用いて課題分析図を作成・保存すると、Moodle のデータベースに課題分析図の情報が保存される。

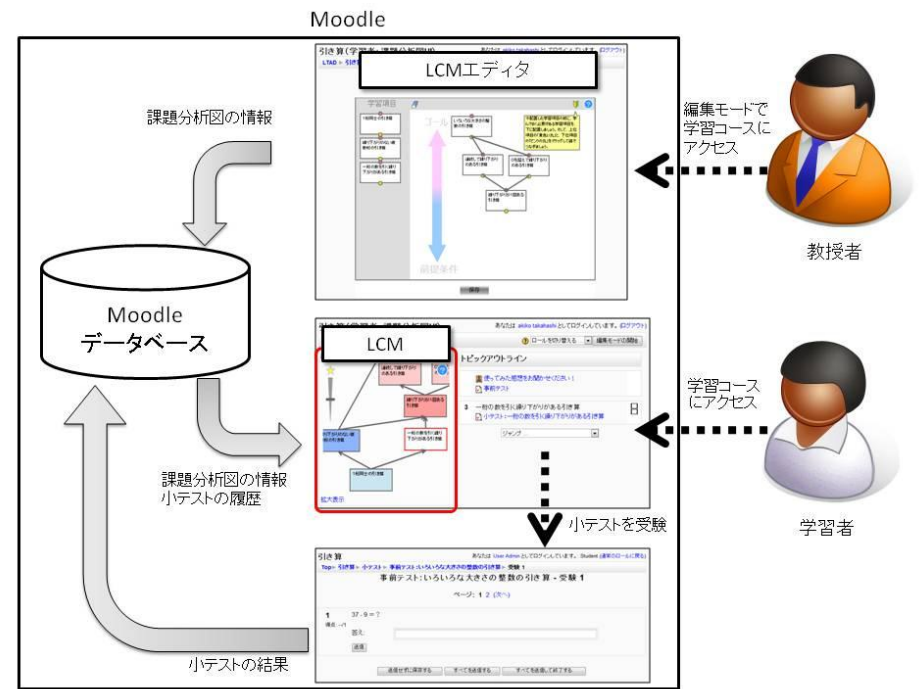


図 2 利用の流れ

一方、学習者は課題分析図作成済みの学習コースのトップページにアクセスすると、ブロック領域に LCM が表示される。LCM 内の課題分析図から、目的の学習項目をクリックすると、該当する学習項目のコンテンツ一覧が表示される。テキストページなどのコンテンツで学習した後、コンテンツの中から小テストを選択し、受験すると、小テストの結果は Moodle データベースに保存される。小テスト受験後に再度学習コースのトップページに戻ると、小テストの成績に応じて LCM 内に表示される課題分析図の学習項目の色が変化する。学習者は課題分析図の構造と色で示された成績情報を参考に、学習内容の選択、コンテンツによる学習、小テストを繰り返すことになる。

LCM および LCM エディタの主要機能については、次項で詳しく述べる。

### 2.3 学習内容選択支援ツール「LCM」の概要

LCM[1]は、1980年代に米国で開発されたインストラクションシステムである TICIT[5]の MAP を参考に、課題分析図で構造を提示し、実際の選択は学習者にゆだねるものとした。学習者が構造情報を参考に自由な順序で学習項目を選択しやすくなるように、Moodle の学習コース内に LCM を設置し、各セクション間の構造関係を表す課題分析図を示すこととした。

TICIT では専用キーボードの MAP ボタンを押したときのみ画面に MAP が表示されたが、LCM は学習者が常に学習内容の構造を意識することを狙い、Moodle のブロックモジュールとして開発した。これにより、課題分析図が学習コース内に常時表示され、図と選択した学習項目（セクション）のコンテンツを一画面の中に並べて表示できる。Moodle の学習コースに表示させた LCM を図 3 に示す。

LCM 内の図の各学習項目は、進捗状況に応じて 5 色で表示した。現段階の LCM は知的技能を対象としており、課題分析図は階層構造を有しているもので、図の下のほうが前提条件となる学習項目である。上に進むにしたがって、下位の学習項目の内容も含んだ、より応用的な学習項目となる。よって、ある学習項目が未修得状態（赤色）となった場合、上位の学習項目の問題は解けないとみなせるので、システム側で自動的に未修得状態（薄い赤色）と判定した。しかし、未修得状態の学習項目でもいくらか知識のある学習者のために、選択はできるようにした。同様に、ある学習項目が修得状態（青色）となった場合、下位の学習項目も含んでいることから、システム側で自動的に修得状態（薄い青色）と判定した。なお、修得状態判定に用いる活動モジュールとして、Moodle の小テストモジュールを用いた。各セクションに 1 つずつの小テストモジュールを配置することで、各小テストの成績データをデータベースから取得して、あらかじめ設定しておいた合格基準を満たしている場合に該当の学習項目を修得状態と判定した。

また、学習者が LCM 内のアドバイスボタンを押したとき、学習順序のアドバイスを提示する。LCM は学習者が学習内容を選択する際に必要最小限の支援をすることとし、学習者の進捗に合わせてシステムが自動的にアドバイスを提示する機能は実装し

ていない。

さらに、操作性を向上させるために、以下の機能を実装した。

- ・図内の学習項目をクリックすると、該当する学習項目のコンテンツが表示される。このとき、LCM の表示領域が非表示にはならない。
- ・ドラッグ操作で図全体を動かすことができる。
- ・スライダバーを上下することで図の拡大・縮小表示ができる。
- ・LCM の表示領域をサイズ変更できる。
- ・図内の学習項目にマウスポインタを重ねると、詳細情報（現在の点数など）が表示される。

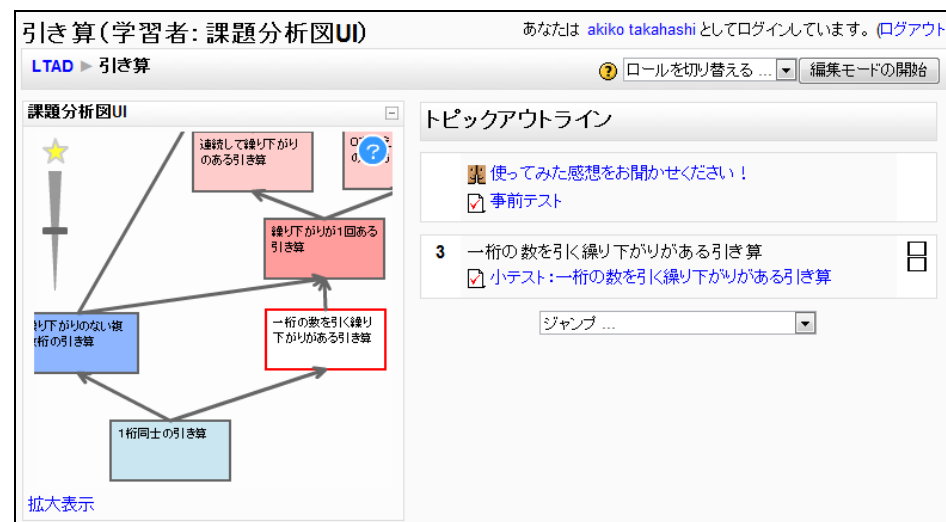


図 3 ブロック化した LCM (高橋ほか 2011[1] 図 3)

### 2.4 課題分析図作成ツール「LCM エディタ」の概要

教授者用の LCM エディタ[2] (図 4) は、UNIKIDS[6]などの先行研究を参考に GUI を実装した。GUI にすることで、教授者が試行錯誤しながら課題分析図を作成しやすくと考えられる。

LCM エディタの画面の左側には、各セクションの Summary 情報の一部を取得し、セクションの一覧が表示される。ユーザは、ドラッグ&ドロップの操作で、セクションを編集領域に移動する。最終目標となるセクションを最上位に置き、前提条件となるより基礎的なセクションを下位に配置し、線をつないでいく。なお、今回はあらか

じめ作成済みのコースに、あとから LCM エディタを導入することを想定した。Summary 情報の入力や小テストの作成は、LCM エディタではなく、Moodle のデフォルトの機能を用いて行うことになる。

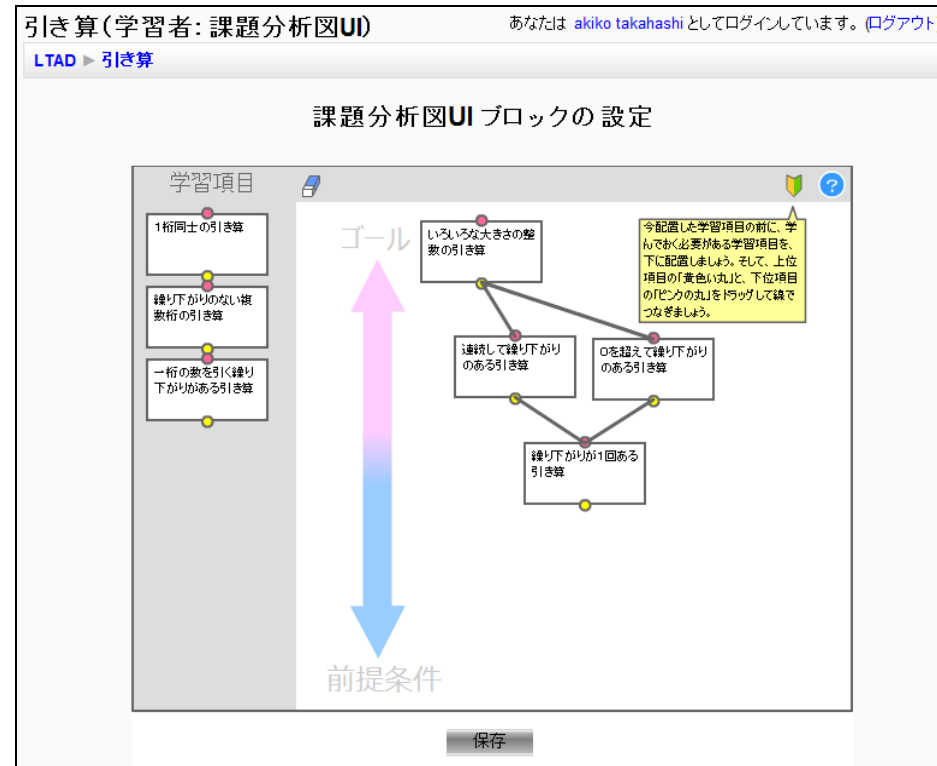


図4 LCM エディタ (高橋ほか 2012[2] 図3)

一方、GUI を用いて操作性を高めても、課題分析図の意味を理解していなければ適切な図を作成することはできない。そこで、課題分析図の作成方法を知らない作成者のために、アドバイス機能を2つ付加した。1点目のアドバイス機能は、自動メッセージである。最初に LCM エディタを表示した際（課題分析図が未作成の場合）は、「最終的なゴールとなる、最後に学ぶべき学習項目を編集領域にドラッグしましょう」というメッセージが表示される。利用者が学習項目を1つ以上配置すると、「今配置

した学習項目を学習項目の前に学んでおく必要がある学習項目を下に配置しましょう」といったメッセージが自動的に表示される。

2点目のアドバイス機能として、ヘルプボタンを設置した。利用者がヘルプボタンをクリックすることで、課題分析図の意味や作成方法の解説、課題分析図の例を含むヘルプ画面が別ウィンドウで表示される。

### 3. 実装

図2における Moodle 内の内部構成を図5に示す。データベース内の白抜き四角形は新規追加テーブルを意味し、灰色の四角形は既存テーブルを意味する。また、実線は主なデータの流れを意味している。システムの詳細は次項で述べる。

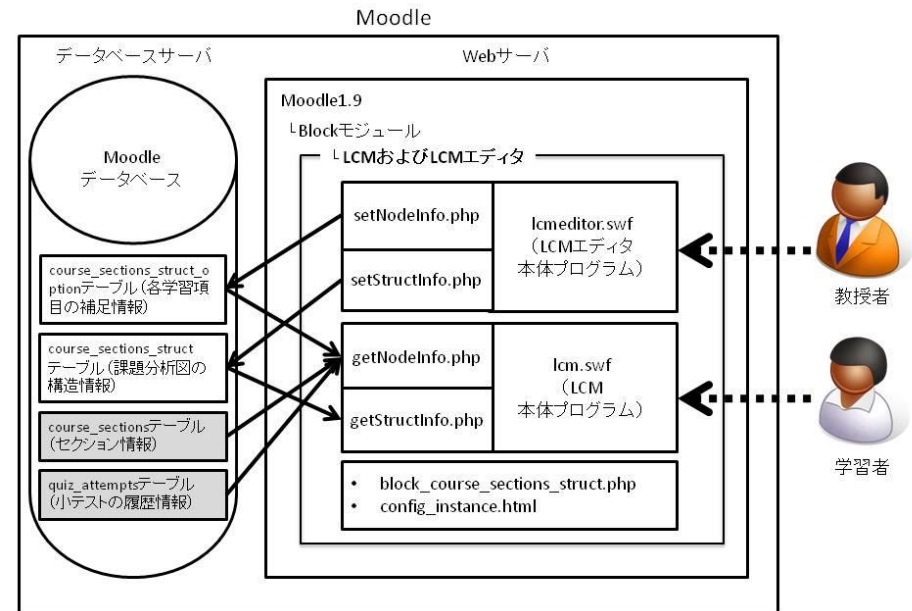


図5 システム構成図

### 3.1 開発環境・動作環境

LCM および LCM エディタの本体プログラムは、Adobe Flash8 および ActionScript2.0 で開発した。各学習項目の補足情報や課題分析図の構造情報の入出力といった Moodle との連携部分は PHP を用いた。

動作環境としては、Moodle1.9 を対象とした。データベース構造が異なるため、現在のところ Moodle2.0 には対応していない。利用者のブラウザは、Flash Player8 以上が動作するブラウザを想定した。

### 3.2 データベース構造

Moodle の既存データベースのうち、主に参照しているテーブルは、course\_sections テーブルと、quiz\_attempts テーブルである。course\_sections テーブルからは、id フィールド (Moodle のコース内のセクション ID)、course フィールド (Moodle 内で使用されるコース ID)、summary フィールド (セクションの概要) の値を取得している。quiz\_attempts テーブルからは、学習者の各小テスト受験履歴から最新の成績情報を取得している。

さらに、Moodle の既存データベースに 2 つのテーブルを追加した。

1 つは課題分析図内の各学習項目の補足情報を保持する course\_sections\_struct\_option テーブルである。このテーブルのフィールドは、id、course、sectionid、x、y、w、h、criteria の 8 つとした。id はデータの識別子である。course には、Moodle 内で使用されるコース ID を意味する course\_sections テーブルの course フィールドの値を保持する。sectionid には、学習コース内の各セクション ID を意味する course\_sections テーブルの ID フィールドの値を保持する。x および y は課題分析図内の学習項目を表す四角形的位置情報 (x 座標、y 座標)、w は四角形の幅、h は高さ、criteria は小テストの合格基準 (デフォルトは 100%) である。

もう 1 つは course\_sections\_struct テーブルである。このテーブルは、セクション間の構造情報 (学習項目をつなぐ矢印の情報) を保持するものである。データの識別子である id のほかに、course、start、end のフィールドを持つ。course には、前述の course\_sections\_struct\_option テーブルと同様に Moodle 内で使用されるコース ID を保持する。start には上位セクションの sectionid、end には下位セクションの sectionid を保持する。

### 3.3 ブロックモジュール用プログラム

LCM および LCM エディタは Moodle のブロックモジュールのため、最低限必要なファイルとして block\_course\_sections\_struct.php および config\_instance.html を作成した。

block\_course\_sections\_struct.php は既存の block\_base クラスを継承しており、ブロックコンテンツとして lcm.swf ファイル (LCM 本体プログラム) を読み込む。また、編集モード時には config\_instance.html が読み込まれることで、編集画面に lcmeditor.swf

(LCM エディタ本体プログラム) が表示される。

### 3.4 LCM の内部構成

block\_course\_sections\_struct.php は、lcm.swf ファイル (LCM 本体プログラム) 呼び出し時に、「コース ID」と学習者が選択した学習項目に対応した「セクション ID」を送信する。lcm.swf は、受け取った「コース ID」を getNodeInfo.php に送信する。getNodeInfo.php は Moodle データベース内の course\_sections テーブル、quiz\_attempts テーブル、course\_sections\_struct\_option テーブルから学習項目 (セクション) 情報と、各学習項目に対応する小テストの成績情報を取得し、その結果を lcm.swf に返す。以上の送受信が完了後、lcm.swf はコース ID を getStructInfo.php へ送信する。getStructInfo.php は Moodle データベース内の course\_sections\_struct テーブルからセクション間の構造情報を取得し、lcm.swf に返す。lcm.swf は、取得した学習項目 (セクション) 情報、成績情報、構造情報に基づき、LCM 内に色分けされた課題分析図を生成する。また最初に block\_course\_sections\_struct.php から受け取った「セクション ID」を基に、学習者が選択した学習項目を強調表示する。なお、LCM ブロックの隣に学習者が選択したセクションだけを表示させる機能は、Moodle の標準機能であるコースセクションの非表示機能を利用しているため、新規開発はしていない。

### 3.5 LCM エディタの内部構成

config\_instance.html は lcmeditor.swf (LCM エディタ本体プログラム) 呼び出し時に「コース ID」を送信する。lcmeditor.swf は、getNodeInfo.php および getStructInfo.php に問い合わせ、すでに課題分析図作成済みの場合は、取得した情報に基づいて、エディタの編集領域に課題分析図を生成する。課題分析図が未作成の場合は、course\_sections テーブルの summary 情報に基づいて、エディタ画面の左側にセクション一覧を表示する。教授者が課題分析図を作成し、保存ボタンをクリックすると、setNodeInfo.php および setStructInfo.php が呼び出される。setNodeInfo.php は course\_sections\_struct\_option テーブルに各学習項目の位置情報などを保存する。setStructInfo.php は各学習項目間の構造情報 (課題分析図内の矢印の情報) を保存する。

## 4. 他 LMS への実装の検討

Moodle のブロックモジュールとして開発した LCM および LCM エディタを他の LMS へ実装する際の課題について考える。ここでは、LMS に備えられている必要がある要件と、LCM および LCM エディタにおいて改修の必要がある点に分けて検討する。

### 4.1 LMS 側の要件

1 点目は、学習コース内でセクション構造を持つことが必要となる。Moodle の学習コースは、セクションという単位で小分けにし、1 つのセクション内にテキストや小

テストなどの複数コンテンツを保持することが可能であった。そのため、学習コース内の構造関係を課題分析図で表現するという事は、セクション間の構造関係を明確にすることを意味する。もし学習コース内にセクション構造を持っていない LMS の場合には、LMS 自体の改修が見込まれるため、LCM および LCM エディタの実装は困難であろう。

2点目の要件は、セクションごとの評価が可能であることである。LCM は、各セクションに1つずつ配置した小テストモジュールを修得状態判定に用いている。セクションに関連付けた評価機能を持たない LMS の場合、LCM の実装はできない。

3点目の要件は、既存データベースの構造が明確であり、なおかつアクセス権限が与えられていることである。既存のデータベース構造が非公開で、データの読み書きが困難な市販 LMS などには、LCM および LCM エディタの実装は難しい。

4点目の要件は、モジュールによる機能拡張が設計されていることである。モジュールやプラグインといった機能拡張が設計されていない場合、LCM および LCM エディタの実装は困難であることが予想される。

## 4.2 LCM/LCM エディタの改修

4.1 で述べた LMS 側の要件が整っている場合でも、LCM および LCM エディタの一部のプログラムを修正する必要がある。

1点目の修正箇所は、データ入出力スクリプトである。本研究および関連研究における開発物の主要プログラムは、LCM 本体プログラムと LCM エディタ本体プログラムである。LCM 本体プログラムは、取得したデータに基づく課題分析図の生成および学習者に対する学習内容選択時のユーザインターフェース機能を担っている。また LCM エディタ本体プログラムは教授者に対する課題分析図の編集機能だけを有している。つまり、別ファイルとして切り分けているデータ入出力スクリプト (getNodeInfo.php, getStructInfo.php, setNodeInfo.php, setStructInfo.php) を、移植先の LMS のデータベース構造およびデータベース接続ライブラリの仕様に沿って開発する必要がある。

2点目の修正箇所は、ブロックモジュール用プログラム (block\_course\_sections\_struct.php および config\_instance.html) である。1点目と同様に、移植先の LMS の仕様に準じてモジュールプログラムを開発する必要がある。

## 5. おわりに

本研究では、Moodle のモジュールとして開発した LCM および LCM エディタについて、他 LMS への実装可能性を検討した。その結果、LMS 側の要件として「学習コース内でセクション構造を持つこと」「セクションごとの評価が可能であること」「既存データベースの構造が明確であり、なおかつアクセス権限があること」「モジュール

による機能拡張が設計されていること」が必要であることが分かった。また、以上の要件が備わっている場合、移植先の LMS の仕様に対応したデータ入出力スクリプトとモジュールプログラムの開発が必要である。

## 参考文献

- 1) 高橋暁子, 喜多敏博, 中野 裕司, 市川 尚, 鈴木 克明: 課題分析図を用いた学習内容選択支援ツールの開発 -Moodle ブロックによる学習者向け機能の実装-, 日本教育工学会論文誌, Vol35, No.1, pp.17-24(2011)
- 2) 高橋暁子, 喜多敏博, 中野裕司, 鈴木 克明: 教授者用の課題分析図作成ツールの開発 -Moodle への実装と評価, 教育システム情報学会誌(2012 掲載予定)
- 3) 鈴木克明: 教材設計マニュアル, 北大路書房(2002)
- 4) ガニエ, ウェイジャー, ゴラス, ケラー (著) 鈴木克明, 岩崎信 (監訳): インストラクショナルデザインの原理, 北大路書房 (2007)
- 5) MERRILL, M.D., *Leaner Control in Computer Based Learning*, Computers & Education vol.4, pp.77-95(1980)
- 6) 右近豊: “実践的インストラクショナル・デザイン技法および支援ツール: “UNIKIDS””, 情報処理学会研究報告, コンピュータと教育研究会報告, 2001(122), pp.1-8 (2001)