

LTI を利用した IT 基礎教育支援ツール max+の開発

max+: A Learning Tool for IT education using LTI [SEP]

藤井 聡一郎¹⁾
Soichiro Fujii

¹⁾ 法政大学情報メディア教育研究センター

I have developed a learning tool for programming named “max+”. It has been developed as a Web Application and can connect to various LMSs using IMS Learning Tools Interoperability (LTI). It provides two kinds of quiz for learning programming. One of those is multiple-choice questions to check basic topics. The other is questions which require to submit source codes. max+ verifies those automatically and returns results immediately. In this paper I show functions of max+, its architecture, and how to set up a whole system on our servers.

Keywords : LMS, Sakai, LTI, SPOC, e-Learning

1. 研究の背景

近年の教育現場では IT の導入が進んでおり、多くの大学で Sakai^[1]や Moodle^[2]などの LMS(Learning Management System)が導入されている。これらの LMS はシステム上での課題管理や簡単なクイズなどの教育支援のための汎用的な機能を持つが、それらの機能だけでは実現できない特定の分野の授業に特化した拡張機能(Small Private Online Course: SPOC)の需要は多い。しかしながらそのような拡張機能は対象となる授業や利用している LMS ごとに異なるものを開発しなければならず、その開発にかかるコストが課題となっている。

2. 研究の概要

本研究の目的は LTI(Learning Tools Interoperability)^[3]に対応した SPOC 向けの拡張機能の開発と、その効果的な開発手法の確立である。LTI とは IMS Global^[4]の定める LMS と拡張機能を連携させるための標準規格で、この規格に準拠した拡張機能を開発することで LTI に対応するすべての LMS に組み込むことが可能となる。近年では Moodle や

Sakai, Blackboard^[5]など多くの LMS や拡張機能が LTI に対応しており、拡張機能に LTI を採用することによって LMS ごとの個別開発のコストを削減することが可能である。

本研究で開発した拡張機能の名称は max+と名づけた。max+では Java プログラミング教育を支援の対象としており、JUnit^[6]を用いた提出プログラムの自動テストなどの機能を持つ。これらの機能は Web アプリケーションの形式で提供され、LTI を用いて LMS と連携して動作する。機能の詳細と実装についてはそれぞれ 3 章、4 章で述べる。

本研究は本年度から開始し、本年度は学内で展開するためのシステムの開発を行った。来年度以降には学内での実証実験や、学外への展開や成果物の公開などを計画している。

本論文では、開発した拡張機能の機能やアーキテクチャ、そこで利用した技術や LMS と拡張機能で構成される来年度の実験のための運用環境についての説明を行う。

3. 本システムの概要

本研究では LTI に対応した SPOC 向けの教育支援

システム max+を実装した。このシステムは Java プログラミングの教育支援を目的としており、プログラミングを学習するための様々な機能を提供する。この章ではその機能について説明する。

max+はプログラミング学習のための課題を提供する機能を持つ。課題の種類は下記の2つである。

- (1) HTML フォームを用いた選択問題などのクイズ (クイズ問題)
- (2) プログラムのコードを記述させチェックさせる問題 (プログラム問題)

参考までにプログラミング問題の画面を図.1 に示す。



図.1 プログラミング問題の画面
Fig.1 Programming Exercise.

いずれの課題形式でも学生が回答を提出した後にサーバ側で自動的に回答のチェックを行い提出した学生へ即座にフィードバックを返すようになっている。この機能により教員は採点の手間を省くことができ、学生は教員の採点を待つことなく自分の解答の正誤を知り、学習を継続することができる。

(1)の回答チェック処理は HTML フォームに入力された値と予め登録しておいた解答データをマッチングして行っている。解答は正規表現で記述することができる。

(2)の回答チェック処理は予め用意された JUnit テストを用いて提出プログラムを検査することで実現している。問題作成者は JUnit のテスト形式で問題に対する回答チェックを記述する。JUnit を用いることで柔軟な回答チェックが可能となっている。

問題の追加には以下の2つの手順が必要となる。

1. 問題ページの追加
2. 解答データの登録

1の問題ページの追加は WordPress^[7]のページ追加機能を用いて行う。max+のフロントエンドはカスタマイズされた WordPress を用いて構成されており、問題文の作成にはそのコンテンツ管理機能を用いることが可能である。

2の解答データの登録はバックエンドのデータベースに対して回答チェック用データを登録する必要がある。プログラム問題に関しては登録専用の Web ページ(図.2)を用意しており、そこからチェック用の JUnit テストを記述したソースコードをアップロードすることが可能となっている。

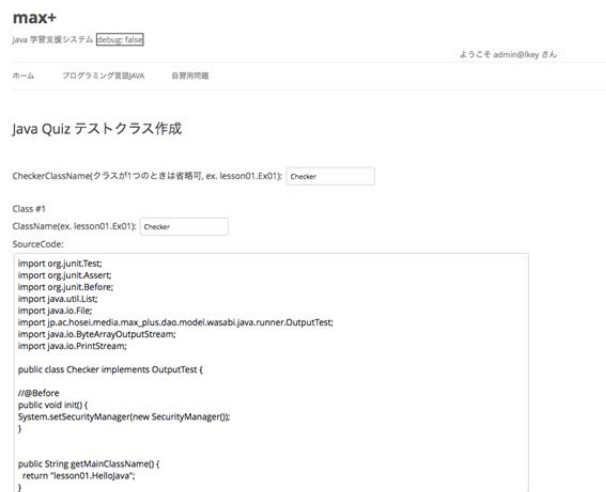


図.2 プログラミング問題編集画面
Fig.2 Edit Page of Programming Exercise

max+では学習状況の確認のために下記の情報を保存している。Web 上からこれらの情報を閲覧する機能は現段階では未実装である。

- 課題の提出内容、提出時間、回答の正否
- ページを開いた日時、閉じた日時

ページの開閉日時の取得には JavaScript を用いているが、その特性上の問題で閉じるイベントがうまく取得できないケースが多く見受けられた。

4. 本システムの実装

本システムはユーザへの画面を提供するフロントエンドと回答チェックや学習ログの蓄積などのシ

システムのロジック部分を担当するバックエンドの 2 つから構成されている。ここではそれぞれの実装について説明する。

4.1. フロントエンドの実装

本システムのフロントエンドは WordPress をカスタマイズして実装されている。説明文や問題文などの画面の提供は WordPress のコンテンツ管理機能を用いて行い、回答チェックの部分はそれらのページから Ajax を用いてバックエンドの Web サービス形式で提供される機能を利用する形で実装している。Web サービスとの Ajax 通信には WordPress 上に通信のための JavaScript を埋め込むことで実現している。具体的にはカスタムテーマやテンプレートなどを作成し、その内部で JQuery^[8]のライブラリを用いてページ内のフォーム要素と Ajax 通信のための JavaScript コードの関連付けを行っている。

4.2. バックエンドの実装

本システムのバックエンドは Web サービス形式で各種機能を提供する。データの形式は JSON を採用しており、フロントエンドと JavaScript などを用いて連携することを想定している。

システムの実装には Java Servlet を用いており、Web サービスの仕様である JAX-RS^[9]を実装したライブラリである Apache CXF^[10]を利用している。その他の主な使用ライブラリは DI のための Spring Framework^[11]や OR マッピングのための Hibernate^[12]を利用している。

提出プログラムの自動チェックには JUnit を利用しており、一般的な JUnit を用いたテストの記述方式を用いた回答チェックの作成が可能となっている。

回答チェック時には提出プログラムを実行しなければならないため、セキュリティ面での対策が必要となる。このシステムでは、提出プログラムの実行を別プロセスで行い、Java のポリシーファイルで実行可能な機能に制限をかけることで不正なコードが実行されないよう対策をとっている。

現段階では本システムで対応している LTI のバージョンは 1.0 のみで、LMS 側へデータを戻すような機能はもたず、データはすべて拡張機能側のデータベースへ格納される仕様となっている。LTI への対応には OSCELOT^[13]の提供する LTI 用のライブラリを用いた。このライブラリは Blackboard 用に実装されている部分が多く、本システムの構成向けに大幅

な修正を行った。

4.3. フロントエンドとバックエンドの連携

LTI1.0 では LMS から LTI ツールの起動リンクをクリックし、ツールの画面をリダイレクトまたはポップアップで表示するのが主な連携方式である。本システムでは LMS からリンクをクリックするとバックエンドのサブレットヘリダイレクトし、そこからフロントエンドの WordPress ヘリダイレクトするという手順でツールの起動を行っている。起動後は起動時に受け取ったユーザ情報を元にバックエンドがセッションを管理し、フロントエンドとバックエンドが Ajax で連携しながら動作する。(図.3)

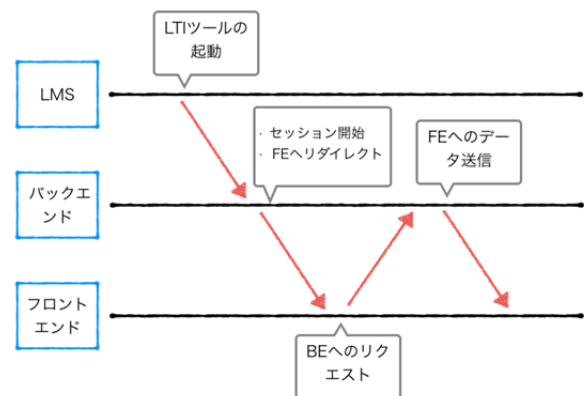


図.3 システムの動作の流れ

Fig.3 A flow of the system

5. 運用環境の構築

本システムの運用環境の構築には情報メディア教育研究センターのマルチメディアサーバを使用した。利用したサーバは mm15 と mm18 の 2 台で OS は両方共 CentOS5.7 である。それぞれのサーバの詳細は以下のとおりである。

mm15:

max+と連携するための Sakai 用のサーバである。Sakai を動作させるために Tomcat と MySQL を利用している。Sakai のバージョンは 10.1 で Sakai の External Tools にて mm18 に配備した max+との連携の設定を行っている。Sakai のユーザ認証は法政大学統合認証 LDAP との連携により統合認証アカウントを用いて行うことが可能である。

mm18:

LTIに対応した拡張機能である max+用のサーバである。使用しているソフトウェアはフロントエンドの WordPress を動作させるための Apache とバックエンド Web サービス用の Tomcat, 両者で利用している MySQL である。Apache と Tomcat は別プロセスで動作しているが Apache Jserv Protocol(AJP)を用いて同一ポートからアクセスできるように設定している。

システムの全体構成を図.4 に示す。

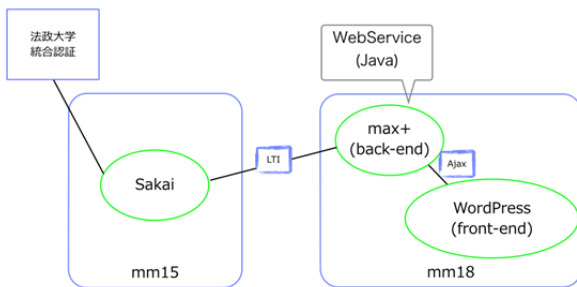


図.4 システムの全体構成

Fig.4 A system structure

6. 今後の展望

今年度のプロジェクトではシステムの実装を主にを行った。来年度には授業内での実証実験を予定しており、実際に使用した際のフィードバックを元にシステムの改良を行う予定である。また、外部公開用の学習コンテンツも制作する予定であり、システムやコンテンツの Open Source Software(OSS)や Open Educational Resource(OER)としての公開や、クラウド上に配備したシステムを Web サービスとして提供することを考えている。システムの機能的な面の改良については学習記録取得のための標準規格である Experience API(xAPI)^[14]への対応や、それを用いたラーニングアナリティクス機能の実装、Java 以外のプログラミング言語への対応、LTI バージョン 2 への対応などを考えている。

7. まとめ

本研究では Java プログラミング教育用の LTI ツールの開発を行った。LTI に対応することにより様々な LMS に組み込むことができ、異なる LMS 間での相互運用性が向上した。このような LTI に対応した SPOC 向けツールをコミュニティ間で共有することにより SPOC 向けシステムの課題の一つである開発

コストの問題をクリアすることが可能だろう。

今後は、OSS や OER, クラウド経由での国内外のコミュニティへの成果の公開を目指してシステムの改良やコンテンツの制作などを進める予定である。

参考文献

- [1]Sakai, <https://sakaiproject.org>
- [2]Moodle, <https://moodle.org>
- [3]Learning Tools Interoperability, <http://www.imsglobal.org/toolsinteroperability2.cfm>
- [4]IMS Global, <http://www.imsglobal.org>
- [5]Blackboard, <http://anz.blackboard.com>
- [6]JUnit, <http://junit.org>
- [7]WordPress, <https://wordpress.org>
- [8]jQuery, <https://jquery.com>
- [9]JAX-RS, <https://jax-rs-spec.java.net>
- [10]Apache CXF, <http://cxf.apache.org>
- [11]Spring Framework, <http://projects.spring.io/spring-framework/>
- [12]Hibernate, <http://hibernate.org>
- [13]OSCELOT, <http://www.oscelot.org>
- [14]Experience API, <http://www.adlnet.gov/capabilities/tla/experience-api.htm>