

論文番号 fm2016-01(仮)

LTIに準拠したネットワーク 自己学習機能の提案と実装

15RD093 菅原 良太, 15RD150 沼田 悠貴

指導: 藤本 衡 准教授

提出日: 2018 年 12 月 25 日

概 要

近年、様々な種類の LMS(学習管理システム:Learning Management System) が存在し、また多くの企業や学習機関が LMS を用いることで e ラーニングを活用している。しかし、LMS は学習者の成績や進捗の管理などを行うのが主な役割である。具体的には教材や資料の配信、共有情報技術者にとってネットワーク技術の理解は必要不可欠であるが、ネットワーク技術は知識として学習しても実際にネットワークを構築しなければ

目 次

概要	2
1 はじめに	4
2 関連研究	5
3 LTI	6
4 LTI	6
5 まとめと課題	8
6 まとめと課題	8
7 まとめと課題	8
8 システム概要	9
8.1 Ruby on Rails	9
8.2 UIについて	10
8.3 機能	11
9 まとめと課題	12

1 はじめに

2 関連研究

3 LTI

4 LTI

ここではLTIについて述べさせてもらう。LTI(Learning Tools Interoperability)とは、IMS Global Learning Consortium(以下、IMT と呼ぶ) が、異なるプラットフォーム間(異なる LMS 上)における学習支援ツールの相互運用を可能とする技術に関する企画を策定し、標準化した規格のことである。LTIに準拠することの具体的なイメージとして、次のようなケースを想定することができる。先代の研究によりできた NSF をツール・プロバイダとし、異なる LMS から利用するケース。これにより、LTIに準拠することで Moodle、や Canvas などの異なる LMS 間で NSF との連携を取ることができた。本研究では Moodle、Canvas での起動を行った。

LTIに置ける用語 Tool Provider(ツール・プロバイダ) Tool Provider(ツール・プロバイダ)とは、外部ツールや外部コンテンツのことであり、本研究では NSF がツールプロバイダとなる。

Tool Consumer(ツール・コンシューマ) Tool Consumer(ツール・コンシューマ)とは、ツール・プロバイダから提供されたツールを使用する LMS のことである。ツール・コンシューマは例として、Canvas,Moodle,Sakai,blackboard などがある。本研究では Moodle、Canvas を使用した。

LTIに置ける利点 LTI 化することにより、様々な LMS からログインすることなく LMS の学習支援ツールとして利用することができる。また、学習支援ツールを、異なる LMS に合わせた設計で作らずに済むことも利点としてあげられる。これによりツール製作者はツールの再利用および、ツールの共有を可能とすることがで

きる。

LTI の利用方法

LMS 上で Tool Provider(ツール・プロバイダ)を使用するには、各 LMS 上で外部ツールの設定を変更する必要がある。例として、moodle では外部ツール設定より、ツール名、ツール URL、コンシューマキー、秘密鍵の設定をする必要がある。これらの設定を得て、moodle から Tool Provider(ツール・プロバイダ)を利用することが可能となる。

外部ツール設定

ツール設定

ツール名 * ?

ネットワークシュミレータ

ツールベースURL * ?

http://localhost:3000/home/create

コンシューマ鍵 ?

tette

共有秘密鍵 ?

...

☐ マスク解除

カスタムパラメータ ?

デフォルト起動コンテナ ?

埋め込み (ブロックなし) ↓

[表示を増やす ...](#)

プライバシー

図 1: moodle 外部ツール設定画面

また、Tool Provider(ツール・プロバイダ), Tool Consumer(ツール・コンシューマ)の間では OAuth 1.0 とを使用して認証している。OAuth (オーオース)とは、SNS や Web サービス間で「アクセス権限の認可」を行うためのプロトコルである。これにより、外部ツールへアクセスする際、ユーザ ID とパスワードによる

ネットワークシュミレータ

ダッシュボード ▶ NS ▶ ネットワークシュミレータ ▶ ネットワークシュミレータ

ネットワークシュミレータ



図 2: moodle 外部ツール起動

認証を行わずに外部ツールへのアクセスを行うことを可能にしている。

5 まとめと課題

6 まとめと課題

7 まとめと課題

8 システム概要

8.1 Ruby on Rails

本研究で提案したネットワークシミュレータは、Ruby on Rails を用いて実装されている。Ruby on Rails とは、Ruby で構築された、Web アプリケーションを開発するためのフレームワークである。特徴として MVC アーキテクチャの採用や設定より規約という設計哲学などが挙げられる。 MVC とは「Model」「View」「Controller」の頭文字であり、MVC アーキテクチャとはアプリケーションの構成が以下の図 3 のようになることに由来している。

図 3: MVC アーキテクチャ (いい感じの画像)

8.2 UIについて

UIの基本的な部分は、魚本、大須賀、中村(2018)らの制作したネットワークシミュレータを採用した。この概要を図4に示す。

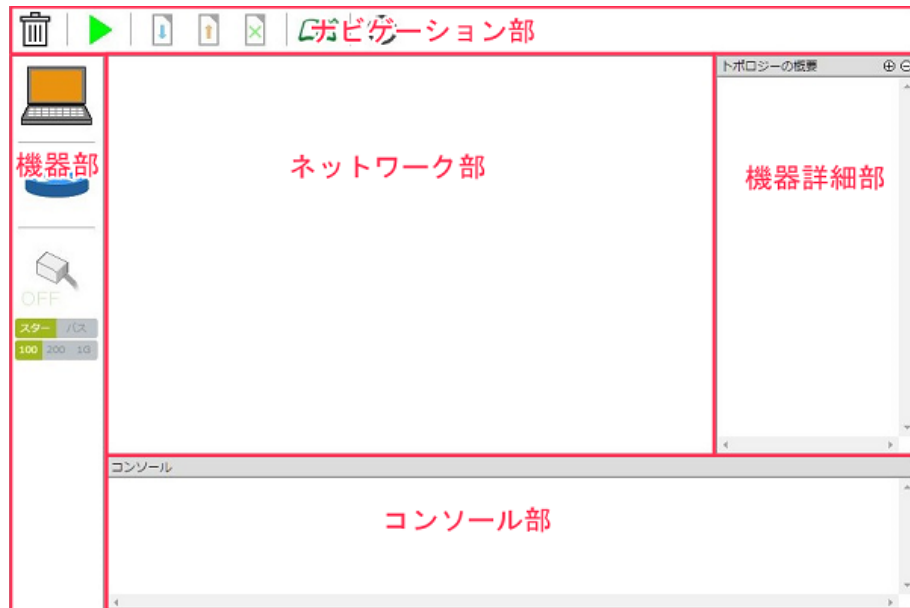


図 4: ネットワークシミュレータ UI(差し替えます)

図4のネットワークシミュレータは、実際にネットワークに関する学習を終えた学生に対しアンケートを行い、9割以上の学生がデザインについて見やすいと答えていた。これにより図4のネットワークシミュレータのUIは変更する必要性がないと判断した。図4は5つの部分に分けられており、機器部、ナビゲーション部、ネットワーク部、機器詳細部、コンソール部となっている。また、図4では自由描画モードと問題演習モードの2つのモードが用意されている。自由描画モードの際、ナビゲーション部ではそれぞれのアイコンをクリックすることでモードの変更、構築したネットワークの正誤の判定、それぞれの機器の詳細情報の確認、

すべての要素の削除を行うことができる。問題演習モードの際は、これに加え練習問題一覧の表示、現在の状況のセーブ、セーブした状態のロード、問題演習モードの終了を行うことができる。機器部では自由描画モードの際に PC やルータをネットワーク部にドロップし、LAN をそれぞれつなげることで自由にネットワークを構築することができる。ネットワーク部では構築されているネットワークのそれぞれの機器に必要な情報を追加する事ができる。これによって正しいネットワークを構築していくことが本ネットワークシミュレータの目的である。機器詳細部はネットワーク部で追加されたそれぞれの機器の情報を確認する部分である。コンソール部は不可能な操作やエラーなどの不具合が起こった場合などにそれぞれの理由や結果などをコンソールとして入力される部分である。

8.3 機能

9 まとめと課題

謝辞

本研究を引き継ぐ際に様々な情報を教えていただいた魚本悠太氏、大須賀旭氏、中村優氏に感謝したいと思います。また、本研究の御指導や実験への協力をして下さいました藤本准教授とシステム評価研究室の皆様に対し、ここに心より深く御礼申し上げます。

参考文献

[1] aaaaa

[2] aa