

仮想計算機を制御する Moodle 仮想演習環境の開発

石井 嘉明[†] 矢野 恭平[†]

富士ソフト株式会社 技術本部 技術開発部 研究開発統括室[†]

1. はじめに

近年, LMS (Learning Management System) は多機能化してきており, テキストやテスト, 課題提出, Wiki, ディスカッションなど, 様々な学習活動を行うことができる基盤となってきた[1]. また, Basic LTI[2]に準拠した LMS であれば, 外部の学習支援ツールの連携も可能である. このように, e ラーニングで提供することが可能な学習コンテンツは非常に広範囲になりつつあるが, LMS において演習や実験を実現するのは困難な状況である.

e ラーニングにおいて技能教育を支援する際は, 学習者によって実機による演習環境を個別に準備し, e ラーニングの演習内容に合わせて演習環境を操作するといった方法や e ラーニングによる演習シミュレータを用意することで実現する方法などが多くとられている. しかし, 前者の場合, 構築作業といった演習内容とは異なる事前作業に多くの時間を要してしまう事が懸念されるだけでなく, 演習履歴が保持されないといった問題がある. また, 後者では, 演習内容に合わせたシミュレータを個別に用意する必要があり, 演習毎に多くのコストを要することになる.

そこで本研究では, これらの問題に対応すべく, e ラーニングによる IT 技術者の技能育成を目指して, LMS から VMM (Virtual Machine Monitor) を制御し, 演習環境を実現するモジュールを開発した. この仮想演習環境では, 学習者の演習開始に合わせ, LMS から仮想計算機の情報収集, 作成などを自動的に行い, 学習者が演習環境を用意することなく, 演習内容に合わせた演習環境が学習者毎に提供されるものである. また, 学習者が演習環境にて操作した学習履歴は LMS 上に保存され, 学習者の復習や教授者による学習管理も可能としている.

Development of the Moodle Virtual Exercise Environment which Controls a Virtual Machine

Yoshiaki Ishii[†], Kyohei Yano[†]

[†] R&D Supervision Section,
Technology Development Department, Technology Division,
FUJISOFT INCORPORATED

2. Moodle による仮想計算機制御機能の開発

本研究では, オープンソースの LMS である Moodle の活動プラグインとして開発を行った. Moodle は, プラグインとして機能拡張することができ, 多くの API を提供している[3]ため, 高い生産性にて開発することができる特徴をもつ. この Moodle からプラグインを介して VMM を制御し, 仮想計算機を作成することで演習環境を提供するよう設計した. 本環境では, VMM は KVM を使用し, PHP 向けに VMM の共通 API を提供する Libvirt-php[4]を用いることで Moodle から仮想計算機を制御している (図 1).



図1 システム構成図

3. Moodle による演習環境の実現

本研究では, 演習環境を実現するために, 仮想計算機を用いているが, 教授者は Moodle の操作のみによって演習環境を用意することができるように, Moodle の活動に連動し, 仮想計算機の制御を実現している.

まず, 教授者は Moodle にて仮想演習の活動を作成する. 仮想演習の活動作成画面 (図 2) では, 演習内容の記載された説明文を登録と合わせて, その演習内容に合致する演習環境をマスターVMのカテゴリから選択する.



図2 活動作成画面

本環境では、MySQL や Java など様々な演習環境を構築したマスターVM をカタログとして複数用意しておく。これによって、教授者は演習環境をそのカタログから選択するだけで、準備は完了する。マスターVM の構築では、Web ターミナルアプリケーションも導入しておく。Moodle からこれと呼び出すことで、演習環境のターミナルを表示する仕組みとした。本開発では、オープンソースの Web ターミナルアプリケーションである WebTTY[5]を使用した。また、カタログに存在しない演習環境が必要な場合は、教授者によって独自に仮想計算機を作成し、VMM 上の指定フォルダに格納しておくだけで、Moodle からは自動的にカタログに追加する仕様としている。

学習者が教授者によって作成された Moodle の活動にアクセスすると、Moodle によって教授者が指定されたマスターVM をコピーオンライトし、起動するよう VMM に指示を行う。その後、演習画面 (図 3) として 1 画面上に演習内容と複製・起動された仮想計算機に接続されたターミナルが表示される。これにより、学習者は演習内容に沿ってターミナル上で演習を行うことができる演習環境が Moodle 上で提供される。

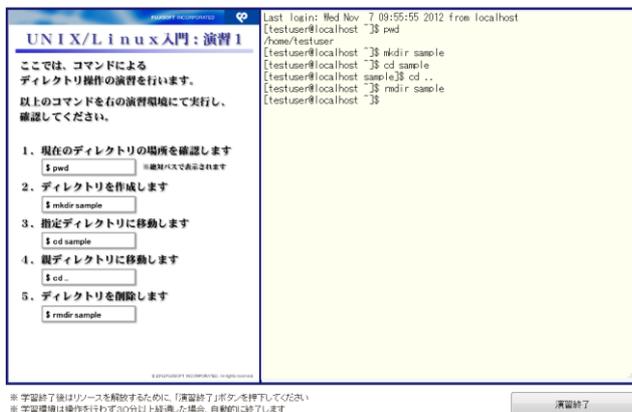


図 3 演習画面

先に述べたとおり、本システムでは、LMS にて演習内容を含めた学習管理を実現するため、学習者がターミナル上で実行したコマンドおよびその実行結果は学習履歴として Moodle 上の DB に格納される。これにより、演習後、学習者の復習や教授者の学習分析をも可能としている。

また、本システムでは、演習後の共有リソースの解放にも対応した。演習後、学習者が手動にてリソースを解放できる他、Moodle は学習者がターミナルへのコマンド入力がある一定時間以上行われていない場合には学習者へ通知し、

その後、仮想計算機を削除することで、リソースの解放を行っている。これにより、無駄のないリソース管理を実現している。

4. まとめと今後の課題

本研究では、IT 技術者の技能育成のため、仮想計算機を用いた演習環境を LMS である Moodle 上に実現した。本システムは教授者や学習者への演習環境の事前準備の負荷を軽減させることを前提とし、Moodle の操作だけで仮想演習環境を実現している。また、演習内容および結果は学習履歴として保持され、リソース管理も行うように開発した。

本研究での取り組みでは、ターミナルを用いた CUI による仮想演習環境を実現したが、今後は GUI による演習にも対応した機能も開発していきたいと考える。

参考文献

- [1] MoodleDocs - Activities, <http://docs.moodle.org/24/en/Activities>
- [2] IMS Global - Learning Tools Interoperability, <http://www.imsglobal.org/lti/>
- [3] MoodleDocs - Core APIs, http://docs.moodle.org/dev/Core_APIs
- [4] Libvirt-php, <http://libvirt.org/php/>
- [5] WebTTY, http://testape.com/webtty_sample.php