Vol.2010-IOT-10 No.11
2010/7/16

Moodle 小テスト時の負荷シミュレーションテスト

王　躍† 小柏　香穂理† 刈谷　丈治†
小河原　加久治††

近年多くの大学では，e ラーニングシステムとして Moodle を構築して運用している．しかし，本格的に Moodle を運用するために，１台のサーバを多数のユーザで同時に利用する場合にどのくらいの負荷に耐えられるのかを把握することは極めて重要である．そこで本研究では，（既存 Moodle 運用環境と同じ）１台のサーバ構成において２種類の HTTP サーバ Apache と Lighttpd に対して，Moodle 小テストを実施する場合の同時アクセス数の負荷テストのシミュレーションを行った．本研究のテスト環境において，PHP アクセラレータを適用したときの結果は，それぞれ，Apache の場合は同時アクセス数が 60～70，Lighttpd の場合は同時アクセス数が 130～140 なら問題なく Moodle 小テストを行うことができる可能性を示唆した．なお，これらの結果は，最大応答時間についての「８秒ルール」のパフォーマンス基準をクリアしている．

Load Test of Moodle Server by Simulating Moodle's Quiz Load

Yue Wang†, Kahori Ogashiwa†, Joji Kariya† and
Kakuji Ogawara††

Moodle is an open source learning management system, widely used all over the world by schools and universities. When Moodle is provide as a formal service, however, the service performance issue is important. For example, how many users can concurrently satisfy with the Moodle service on a single-server system? This paper addresses the issue of Moodle's performance within two different HTTP servers - Apache and Lighttpd. Under the 8-second rule, it is shown that in our test environment with PHP eAccelerator, for a Moodle's Quiz test the number of maximum concurrent users supported by an Apache server is about 60 to 70 , whereas the number of maximum concurrent users supported by a Lighttpd server is about 130 to 140.

1. はじめに

近年，LMS（Learning Management System）としてのMoodleの導入は，世界の大学で進んでいる[1]．日本においてもMoodleによるeラーニングシステムの構築と利用に取り込む大学や高校が増えている．Moodleはオープンソースソフトで誰でも無料で利用できる．しかも，教育支援ツールとしての機能が豊富で比較的簡単に構築できるので，試しに導入して利用されるケースが多いと思われる．しかし，本格的にMoodleを運用して大学の 100 人の講義に利用する場合は，１台のサーバを多数のユーザで同時に利用する場合にどのくらいの負荷に耐えられるのかを把握することは極めて重要である．そこで本研究では，（既存Moodle運用環境と同じ）１台のサーバ構成において２種類のHTTPサーバApacheとLighttpdに対して，Moodle小テストを実施する場合の同時アクセス数の負荷テストのシミュレーションを行った．

2. Moodle 負荷テスト

本研究では，実運用のネットワーク環境に即すため，テストサーバとテストクライアントを学内 LAN の異なるサブネットに接続させた．テストサーバへの同時アクセスは，テストクライアントのマルチスレッドを用いて仮想的に実現した．テストサーバには，あらかじめ必要なテストユーザ人数分のアカウントをローカルに登録してある．また，ユーザ（スレッド）は，平均５秒の間隔（思考時間）で次のページへアクセスしていくと想定した．

2.1 テスト環境

本研究では，次のようなテスト環境構成で実験を行った．

(1) **サーバ構成**：

CPU：Dual-Core AMD Opteron (tm) Processor 1218 (2613.40-MHz K8-class)（２コア）
メモリ：DDRⅡ DIMM 5300 Unbuffered 2GB×4
HDD：SATA 400GB×8（400GB×2 RAID1 ＋ 4 00GB×6 RAID10）
OS：FreeBSD 7.2-RELEASE (amd64)
Apache：2.2.11 (FreeBSD)
Lighttpd：1.4.25 (ssl)
PHP：PHP5.2.9 (fastcgi) with Suhosin-Patch 0.9.7 (cli) Zend Engine v2.2.0
PHP アクセラレータ：eAccelerator v0.9.5.3
RDBMS：MySQL5.0.77（ソケット接続）

†山口大学　大学情報機構　メディア基盤センター
Media and Information Technology Center, Yamaguchi University
††山口大学　理工学研究科
Graduate School of Science and Engineering, Yamaguchi University

ⓒ 2010 Information Processing Society of Japan

Moodle：1.9.7+（※キャッシュタイプ eAccelerator のレコードキャッシュの設定を有効にしている）
LAN：100MB

(2) **クライアント構成**：
CPU：Intel® Core™ i7 860@2.80GHz（４コア）
メモリ：8GB（2.99GB 使用可能）
HDD：SSD40GB＋SATA800GB
OS：Windows 7 Ultimate （32 ビット）
JAVA：version "1.6.0_17"
Apache JMeter：version 2.3.4 r785646
LAN：100MB（※サーバとクライアントは異なるサブネットに接続されている）

2.2 負荷テストツール

本研究での負荷テストは，Apache J Meter2.3.4 を用いて，Moodle の小テストを行う時のユーザー一連の動作を模倣して行う．ただし，Moodle のログイン認証は異なる学生アカウントでのローカル認証を用いている．

次に JMeter テスト計画のツリー構造（**図 1**）に沿って各設定項目を説明する．

- スレッドグループ：スレッド数 ＝10（～150），Ramp-Up 期間= 10 秒，ループ数= 2
- HTTP クッキーマネージャ：Cookie Policy = compatibility
- HTTP リクエスト初期値設定：Web サーバ IP＝ サーバ IP アドレス, ポート番号＝ 80 または 3000（※Web サーバが Apache の場合は，ポート番号＝ 80，Lighttpd の場合は，ポート番号＝ 3000），プロトコル＝ http，コンテンツエンコーディング＝ utf-8，パス＝ Moodle サイトのトップページへのパス
- ガウス乱数タイマのスレッド遅延プロパティ：偏差（ミリ秒）= 300 0.0，遅延時間オフセット定数（ミリ秒）= 5000.0
- 一度だけ実行されるコントローラ：「Moodle サイトのトップページ」と「ログインページ（GET)」と「ログインページ（POST)」との３つの HTTP リクエストから構成されている．なお，ログインページ（POST）のリクエストで送るパラメータは，「username」「password」「testcookies」となる（**図 2**）．「username」と「password」の値は，JMeter の「ユーザーパラメータ」及び「HTTP ユーザーパラメータの変更」で定義する（**図 3**）．すべてのテスト用学生アカウント（username, password）は，JMeter の/bin ディレクトリにあるファイル users.xml に含まれる．
- シンプルコントローラ：基本的に３つの Moodle 小テスト（※２つの多肢選択問題と１つの穴埋め問題）から構成されているが，いくつかのビューページも含まれている．

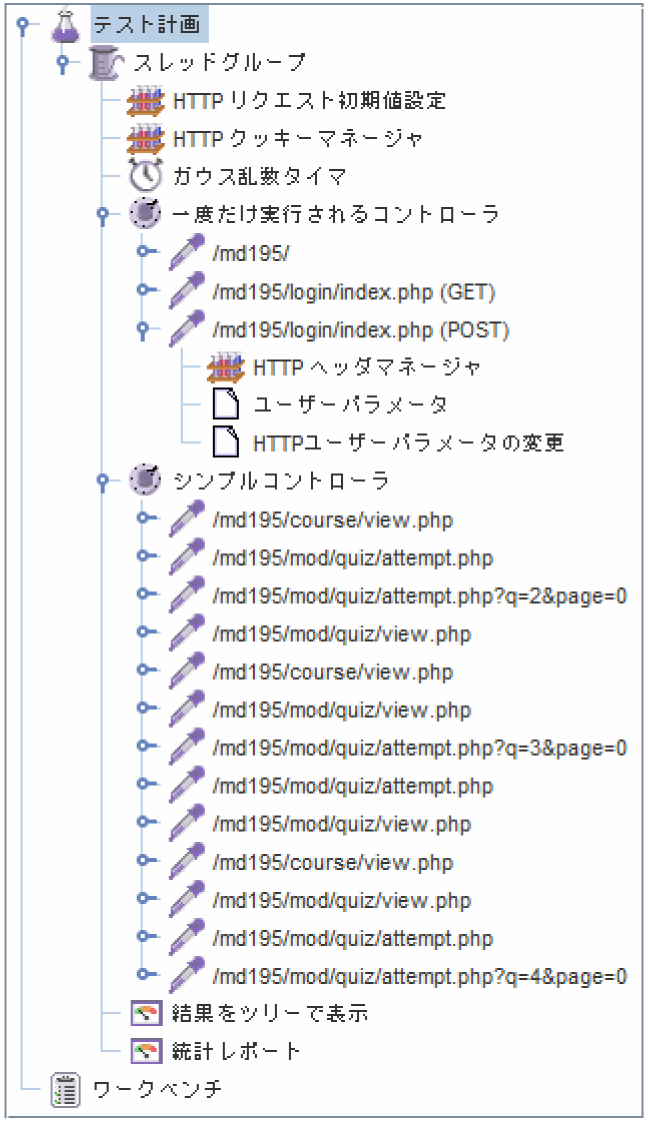
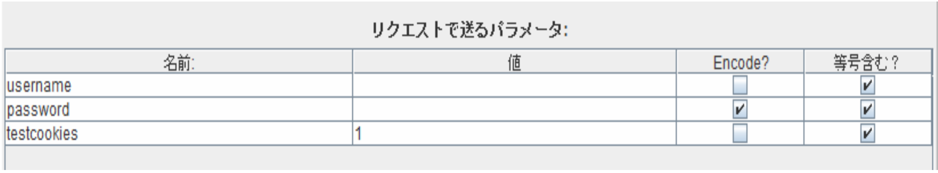


図 1 JM eter テスト計画

ⓒ 2010 Information Processing Society of Japan



リクエストで送るパラメータ:

名前:	値	Encode?	等号含む？
username		☐	☑
password		☑	☑
testcookies	1	☐	☑

図 2　ログインリクエストで送るパラメータ

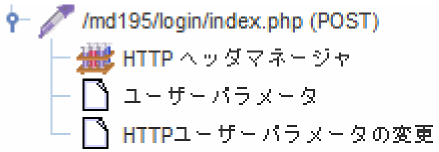


図 3　パラメータの指定

● 統計レポート：JMeter のテスト結果の統計レポート
● 結果をツリーで表示：HTTP リクエストの応答を Web 画面で確認できる．

なお，Moodle小テストのシナリオは，JMeterのHTTPプロキシサーバ機能[2]を用いて自動的に作成したものに基づいて編集した．また，すべてのHTTPリクエストは，「リダイレクトに対応」と「KeepAliveを有効にする」と設定した．

2.3　テスト結果

本研究では，WebサーバとDBサーバのチューニングについては，参考文献[3]を参照している．また，サーバにはPHP高速化するためのアクセラレータeAcceleratorを導入している[4]．Mdooleのパフォーマンスでは，キャッシュタイプをeAccelerator，レコードキャッシュを有効と設定している．Webサーバの最大応答時間については，「８秒ルール」[5]を基準にして，最大応答時間が８秒を超えた時点で，サーバの負荷受容量を超えたと考える．サーバApacheとサーバLighttpdに対して，それぞれ，その負荷受容量を測るため，同時スレッド数を 10 個ずつ増やして測定した．その結果，Apache とLighttpdは，それぞれ，同時スレッド数が 80 と 150 のとき，初めて最大応答時間が８秒を超えた．

Web サーバの最大応答時間は図 4 に示す．Lighttpd に比べて Apache の場合は，最大応答時間の伸び具合が指数関数のような特徴がみられる．

Web サーバのスループットは図 5 に示す．図 5 によれば，Apache のスループットのピーク値は，同時スレッド数が 60～70 ところにある．その後，スループットが低下したので，サーバが既に飽和状態に達していることを意味する．一方，Lighttpd の方は，同時スレッド数が 70 まではほぼ線形増加し，80 以降は緩やかに増加している．

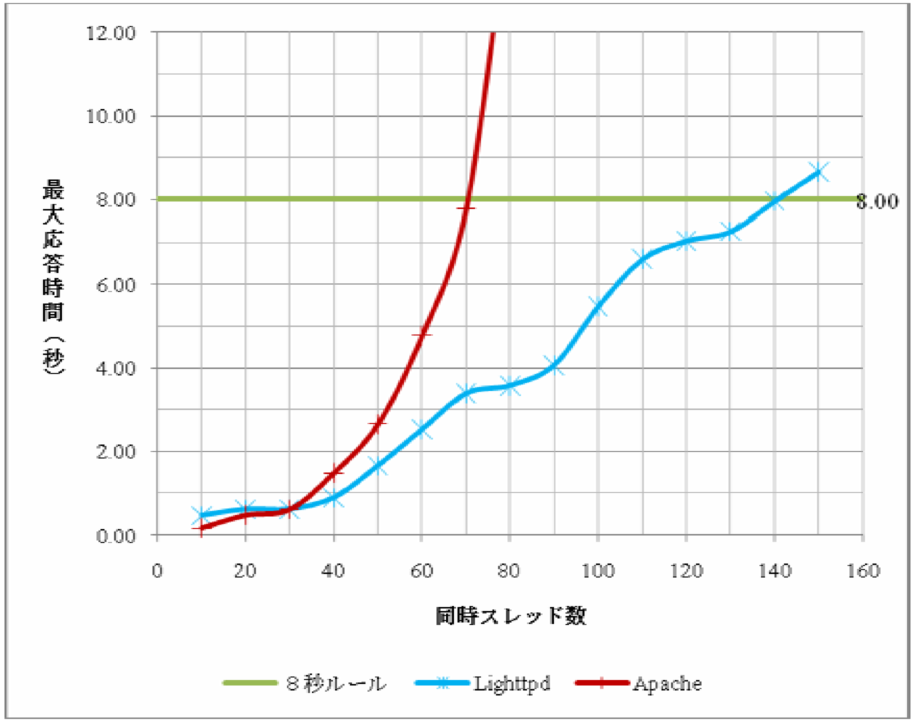


図 4　最大応答時間

Web サーバの CPU_LOAD_AVERAGE の１分間最大値を図 6 に示す．Lighttpd と比べて，Apache の場合は，CPU_LOAD_AVERAGE の１分間最大値の伸び具合に，最大応答時間の場合と同様に，指数関数のような特徴がみられる．また，同時スレッド数が 70 のとき，CPU_LOAD_AVERAGE の１分間最大値が 10 を超えたので過負荷となった．一方，Lighttpd の方の CPU_LOAD_AVERAGE の１分間最大値は，緩やかに増加し，同時スレッド数が 140 になった際も，4 ぐらいに止まった．

3.　考察

本研究では，サーバ１台の構成において，標準ソフトウエアを用いて Moodle 小テスト時の負荷シミュレーションテストを行った．テストに使われた Moodle の小テス

ⓒ 2010 Information Processing Society of Japan

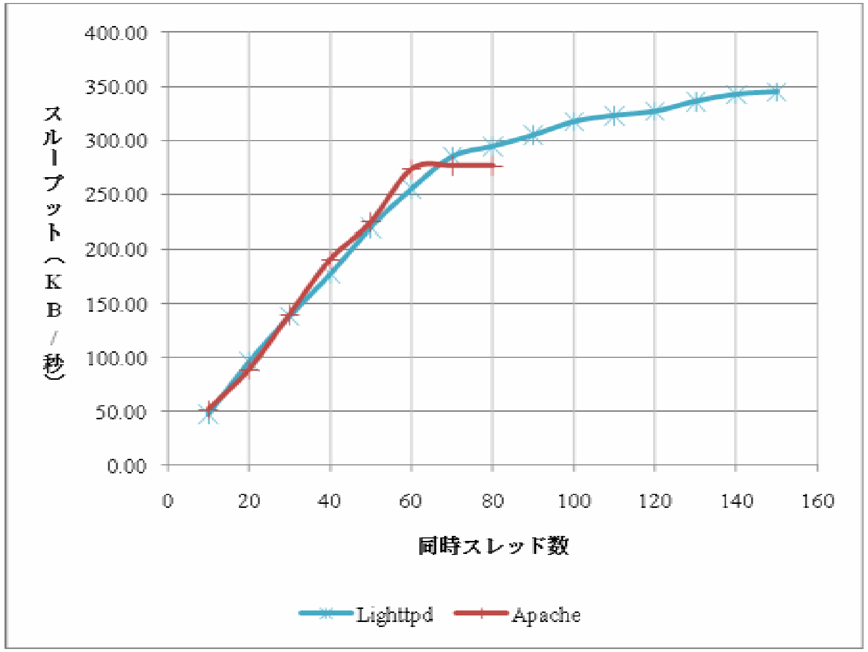


図 5　スループット

ト問題が単純なものであったため，ネットワークの帯域幅がテスト結果への影響はないと考えられる．したがって，本研究でのテスト結果は，主にサーバのハードウエアのスペックとサーバのソフトウエア（OS, DB, PHP, Moodle, Apache or Lighttpd …）の性能に依存する．特に， Apache と Lighttpd とのテスト結果の比較では，負荷受容量（同時スレッド数）に２倍以上の大きな差がみられることになったので，負荷受容量が Web サーバソフトの性能に強く依存していると思われる．

なお，本テストでの Moodle ユーザ認証方法は，Moodle ローカル認証を用いているが，本研究の実験では，Web 応答時間に関しては，Moodle のローカル認証と POP3 認証または LDAP 認証との間の差は，ほとんどないことが確認済みである．

4. まとめと今後の課題

本研究におけるMoodle小テスト時の負荷シミュレーションテストの結果より，１台のサーバ構成について，PHPアクセラレータeAcceleratorを適用した場合，WebサーバApacheとLighttpdは，それぞれ，60～70 と 130～140 の同時アクセスなら問題なく，「８秒ルール」の応答時間の基準に基づいたMoodle小テストを行うことができる可能性を示唆した．ただし，１台クライアントからのアクセスと数十台あるいは百台のクライアントからのアクセスとでは，サーバ負荷に関して全く異なる特性を示す可能性があるので，本研究でのシミュレーション結果は，必ずしも実運用の場合に一致するとは限らない．実際に，１台サーバ構成の実運用環境において，300 人規模のMoodle小テスト（試験）の成功事例は既に報告されている[6][7].

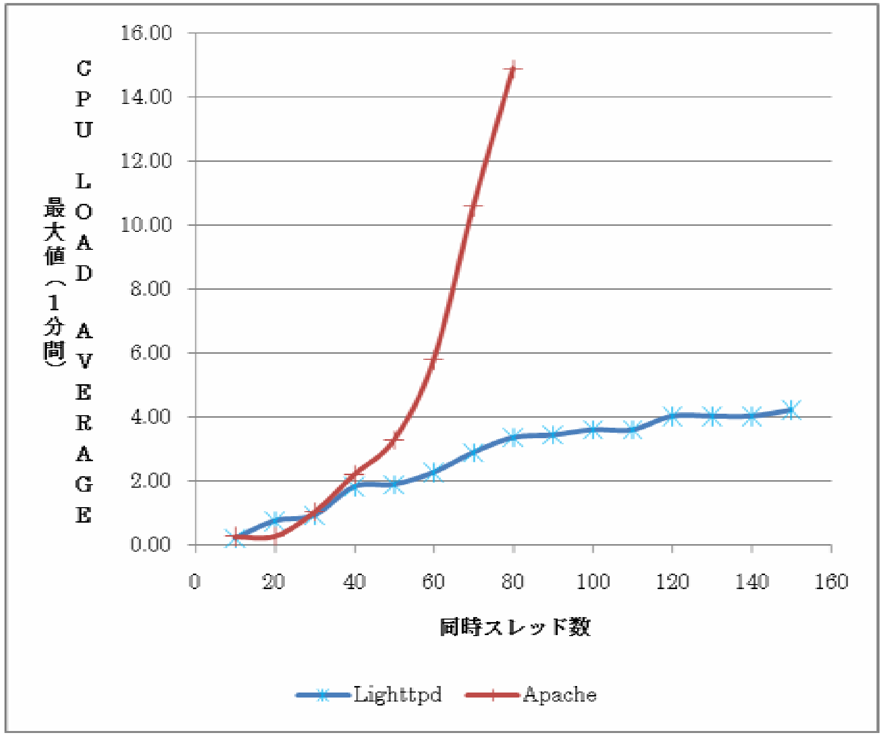


図 6　CP U_LOAD_AVERAGE 最大値

今後の課題は，次のようにあげられる．

- アクセスされる Moodle ページのサイズが大きい場合（例えば，画像やビデオなどを含む SCORM 教材）での負荷テスト

ⓒ 2010 Information Processing Society of Japan

Vol.2010-IOT-10 No.11
2010/7/16

- JMeter のリモートテスト機能を利用して複数の PC からの同時アクセスした際の負荷テスト
- ２台のサーバ（Web サーバ＋DB サーバ）構成における負荷テスト

謝辞　本研究にご協力いただいた山口大学・学情報機構・メディア基盤センターの皆様に，謹んで感謝の意を表する．

参考文献

1) M oodle「Registered moodle sites」http://moodle.org/sites/
2) Apache Jakarta Project「JMeter proxy Step-by-step」
http://jakarta.apache.org/ jmeter/usermanual/jmeter_proxy_step_by_step.pdf
3) M oodle「Performance」http://docs.moodle.org/en/Performance
4) 辻靖彦, 杉山秀則, 平野秋一郎, 小野博: UPO-NET 教材への同時アクセス状況における
Moodle 負荷試験, 日本教育工学会第 25 回全国大会講演論文集, 3a-221-07, pp. 823-824 (2009).
5) チ・チャン・クン「テスト・スクリプト改善のヒント」
http://www.itarchitect.jp/print/?menu3=39302
6) Don Hinkelman「Quiz Handles 300 Simultaneous Users on One Server」
http://moodle.org/mod/forum/ discuss.php?d=68579.
7) 原田寛之（ムードル・ムート函館２０１０）「サーバ管理者による大規模テストの事例報告」
http://vle.c.fun.ac.jp/moot2010/.

ⓒ 2010 Information Processing Society of Japan