

Web パフォーマンス測定プラットフォーム

中野 雄介^{†,††} 上山 憲昭^{†,††} 塩本 公平^{††}

長谷川 剛^{†††} 村田 正幸[†] 宮原 秀夫[†]

†† 日本電信電話株式会社 NTT ネットワーク基盤技術研究所 〒180-8585 東京都武蔵野市緑町 3-9-11

††† 大阪大学サイバーメディアセンター 〒560-0043 大阪府豊中市待兼山町 1-32

† 大阪大学大学院情報科学研究科 〒565-0871 大阪府吹田市山田 2-1

E-mail: †{nakano.yuusuke,kamiyama.noriaki,murata,miyahara}@ist.osaka-u.ac.jp,

††shiomoto.kohei@lab.ntt.co.jp, †††hasegawa@cmc.osaka-u.ac.jp

あらまし 近年, Web パフォーマンスの重要性が注目を集めている. Web パフォーマンスとは, Web ページ上のリンクがクリックされてから, 次の Web ページを構成するオブジェクトがダウンロードされ, 表示が完了するまでの時間である. ユーザは Web パフォーマンスが低い Web ページから離れる傾向にあり, Web パフォーマンスの低下はサービス提供者の収入の低下に直結する. このため, サービス提供者は自身が提供する Web ページのパフォーマンスを測定し, パフォーマンス低下の原因を究明, 改善する必要がある. しかし, このような Web パフォーマンスの低下原因は, ネットワーク環境や端末の性能等, Web ブラウザの動作環境によって変わると考えられる. 本稿では, Web ブラウザの動作環境を変化させ, 且つ, Web パフォーマンスに加えてサーバ・ネットワーク・クライアントそれぞれのパフォーマンスについて測定可能とする, Web パフォーマンス測定プラットフォームを提案する. 提案手法を評価した結果, 提案手法は 959 個の Web ページの 102 のホストでの測定において, 測定にかかる稼働は 30 分程度と少なく, PlanetLab を用いることで, 多様な Web ブラウザの動作環境での測定ができることを確認した. これにより, 測定結果から, Web ページのパフォーマンス改善の方針を検討できるデータを収集することができた.

キーワード Web パフォーマンス, 測定環境

Platform for Web Performance Measurement

Yuusuke NAKANO^{†,††}, Noriaki KAMIYAMA^{†,††}, Kohei SHIOMOTO^{††},

Go HASEGAWA^{†††}, Masayuki MURATA[†], and Hideo MIYAHARA[†]

†† NTT Network Technology Laboratories, NTT Corporation Midori-cho 3-9-11, Musashino-shi, Tokyo, 180-8585 Japan

††† Cybermedia Center, Osaka University 1-32, Machikaneyama, Toyonaka-shi, Osaka, 560-0043 Japan

† Department of Information Science, Osaka University 1-5, Yamadaoka, Suita-shi, Osaka, 565-0871 Japan

E-mail: †{nakano.yuusuke,kamiyama.noriaki,murata,miyahara}@ist.osaka-u.ac.jp,

††shiomoto.kohei@lab.ntt.co.jp, †††hasegawa@cmc.osaka-u.ac.jp

Abstract Web performance is getting important in recent years. Web performance means the time from clicking a link on a web page to finishing showing the web page of the link and low web performance web pages are tend to lose customers. To avoid web performance degradations, service providers need to find the cause of the degradations and improve their web performance. However, web performance depends on web browser running environment such as network environment and calculation resources of clients. In this paper, we propose a platform for web performance measurement which measures not only web performance but also performance of server, network and client with changing web browser running environment. Our experimental result shows that the proposed platform takes 30 minutes to start a measurement of 959 web pages' performance from 102 hosts. In addition, the platform measures web performance in various web browser environments by using PlanetLab. As a result, we obtain data for web performance improvement.

Key words web performance, measurement platform

る。このイベントを契機としてそれぞれのオブジェクトのダウンロードが開始される。図1の例では、サーバXに対するオブジェクトのダウンロードのイベントが発生し、その後、立て続けにサーバAに対する3つのダウンロードのイベントが発生する。このように、オブジェクト間には依存関係があり、その依存関係に従った順番でオブジェクトはダウンロードされる。また、単一のWebサーバから複数のダウンロードを行う際は、一定の並列数に制限される。この例では、並列数は1とされており (Chrome では6に設定されている)、1つのオブジェクトがダウンロードされている間、後のダウンロードはエンキューされ、デキューされるまで待ち、この時間はBlockedと呼ばれる。なお、Blockedの原因には様々なものがあり、文献[3]で紹介されている。その後デキューされ、ダウンロードが開始される。まず、WebサーバのIPアドレスを取得するため、DNSサーバを用いて名前解決する。この時間はDNSと呼ばれ、DNSキャッシュされている場合は、この時間は0となる。WebサーバのIPアドレスが判明すると、Webブラウザはサーバとコネクションを確立し、HTTPリクエストを送信し、HTTPレスポンス受信まで待ち、HTTPレスポンスを受信する。これらの時間はそれぞれConnect, Send, Wait, Receiveと呼ばれる。

このようにして分離された各内訳は、関連する箇所 (サーバ・ネットワーク・クライアント)がある程度決まっているため、オブジェクトのダウンロードの際に最も時間のかかる内訳がわかれば、Webパフォーマンス低下の原因を含む箇所を特定することができる。つまり、Blockedはクライアントに関する時間であり、DNS, Connet, Send, Receiveはネットワークに関する時間である。一方、Waitはサーバでの処理時間と、ネットワークでのRTT両方が含まれる。このため、Waitをサーバとネットワークそれぞれにかかる時間に分離するため、RTTについても測定する必要がある。

なお、onload時間と各オブジェクトのダウンロード時間の内訳はFirefoxやChromeといった近年のWebブラウザによって測定することができる。また、測定結果はHAR(HTTP Archive)[4]と呼ばれるJSONファイルとして出力できる。

以上から、Webパフォーマンス測定プラットフォームは測定データとしてonload時間、Blocked, DNS, Connet, Send, Wait, Receive (HAR形式)、RTTを収集する必要がある。

2.2 Webブラウザの動作環境

Webパフォーマンスの低下原因はWebブラウザの動作環境によって変化する。また、Webページは多様な環境で動作するWebブラウザによって表示される。このため、Webページの提供者は、多様なWebブラウザの動作環境において、自身のWebページのパフォーマンスを測定する必要がある。Webブラウザの動作環境を決定する要素には以下の様なものがある。

ネットワーク内での端末の位置 端末の位置によって、Webパフォーマンスは変化する。これは、回線の性能 (RTTとスループット)と、ダウンロードするオブジェクトとの距離が変化するためである。例えば、端末が南米にある場合、北米のWebサーバが提供するWebページに対するWebパフォーマンスは、端末が北米にある場合と比較して悪くなる。

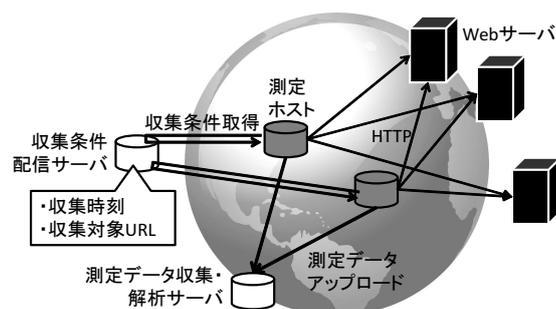


図2 Webパフォーマンス測定プラットフォームの概要

時刻 Webページを表示する時刻や曜日によっては、輻輳等の発生によりWebパフォーマンスが低下することがある。

端末の処理性能 端末の計算リソースが少ない場合は、スワップの発生等によりWebページのレンダリングに時間がかかり、Webパフォーマンスが低下すると考えられる。

以上のような要素の組み合わせにより、Webパフォーマンス測定プラットフォームは多様なWebブラウザの動作環境を実現する必要がある。

3. Webパフォーマンス測定プラットフォームの提案

以上のような要求条件を満たすWebパフォーマンス測定プラットフォームを提案する。図2に提案プラットフォームの概要を示す。提案プラットフォームは世界中に配置された測定ホスト、測定対象のURLと測定開始時刻とを保持・配信する測定条件配信サーバ、測定データ収集・解析サーバから構成される。

今回、測定ホストには、インターネットに接続された世界中のホストを自由に利用できる実験環境である、PlanetLab[5]を用いた。各地のホストにWebブラウザと測定プログラムをインストールし、測定用ホストとした。PlanetLabのホストは、様々な大学や企業が自身の保持するホストマシンを提供することで成り立っている。提供されるホストマシンの処理性能は様々であるため、予め各ホストでベンチマークソフトを走らせ、ホストごとの処理性能を測定しておく。なお、ベンチマークソフトとしては、UnixBenchを用いた。一方、PlanetLabを用いず、独自でホストを配置し利用してもよい。この場合、世界各地に様々な処理性能のホストを配置することになる。

測定条件配信サーバはWebサーバであり、測定対象のURLと測定開始時刻とを保持・配信する。測定ホストにインストールされた測定プログラムは、測定条件配信サーバから、測定対象URLと測定開始時刻を取得し、それらに従ってWebパフォーマンスを測定する。

測定データ収集・解析サーバはFTPサーバであり、各測定ホストで収集されたHARとRTTのデータはFTPを介して測定データ収集・解析サーバにアップロードされる。

3.1 測定ホスト

図3に測定ホストの概要を示す。測定ホストには、予め測定プログラムとFirefoxがインストールされている。測定プログラムはCronなどによって起動される。その後、測定プログラ

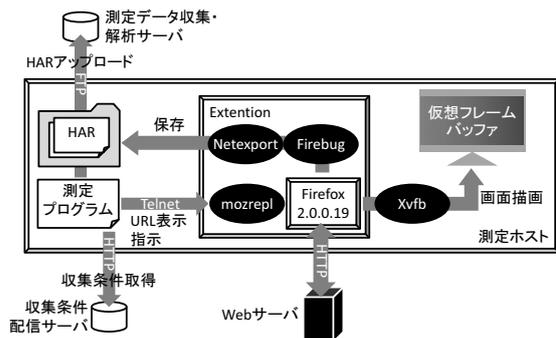


図 3 測定ホストの概要

ムは、測定条件配信サーバから HTTP を介して測定条件を取得¹⁵する。測定条件が公開されていない場合、測定プログラムは測¹⁶定条件が公開されるまで待つ。測定条件には測定開始時刻が記¹⁷載されており、測定プログラムはその時刻まで待機する。なお、¹⁸測定ホストのある国の現地時刻での測定開始時刻まで待機する必要がある。一方、PlanetLab のホストの時刻は全て UTC に¹⁹設定されている。このため、GeoNames [6] が提供するデータを用い、測定ホストの位置情報から現地時刻を特定し、その時刻から待機する時間を算出する。

待機時間が満了すると、測定プログラムは Web ブラウザを起動・操作する。今回は Web ブラウザとして Firefox を用いた。Firefox のバージョンは 2.0.0.19 であり、キャッシュをローカルに保存しない設定としている。測定プログラムは Firefox を Telnet 経由で遠隔操作する Extention である mozrepl を用いて、Firefox に測定条件に含まれる測定対象の URL を表示するよう指示する。なお、Firefox 以外の Web ブラウザを用いる場合、遠隔操作のためのプログラムや、HAR の生成方法について、Web ブラウザにあわせて選定する必要がある。Firefox は該当する URL の Web サーバに対して HTTP リクエストを送信し、その Web ページを表示する。PlanetLab ノードには画面が存在しないため、Xvfb を介して仮想フレームバッファにレンダリング結果を描画することで、実際の Web ブラウザの動作を実現している。PlanetLab 以外の仮想サーバ等を利用する場合も画面がないと考えられるため、同様の手法を用いることができる。HAR の生成には、Web ページのデバッグ等の機能を提供する Extention である Firebug と、Firebug 上で動作する HAR 生成用の Extention である Netexport を用いる。

測定プログラムは HAR が生成されると、生成された HAR を解析し、Web ページを構成するオブジェクトを提供する Web サーバのホスト名を抽出する。その後、各ホストに対して Ping を送信し、RTT を測定し、測定結果をファイルに保存する。このようにして、HAR と RTT の測定結果を生成する。

最後に、生成された HAR ファイルと RTT の測定結果のファイルとを測定データ収集・解析サーバに FTP でアップロードする。このようにして、測定プログラムは、収集対象の全ての URL に対する HAR と RTT とを生成し、アップロードする。

なお、Firefox のプロセス停止などにより、測定不能となることがある。測定プログラムは HAR ファイルの生成状況を監

```

1 {
2   "path_to_har":"/home/test/logs/",
3   "path_to_firefox":"/home/test/firefox",
4   "location_url":"http://osakafnet.com/location.txt",
5   "path_to_firefox_targz":"/home/test/env.tar.gz",
6   "ftp":{
7     "host_name":"osakafnet.com",
8     "port":21,
9     "dir_name":"data/"
10  },
11  "timing":[
12  {
13    "start":"0:0:0",
14    "timeout":30,
15    "target_list_url":"http://www.osakafnet.com/url.txt"
16  },
17  :
18  :
19  }

```

図 4 測定条件の例

視し、HAR が生成できていない場合、Firefox の動作が不完全であると判断し、復旧を図る。復旧には予め作成しておいた、正常に動作する Firefox の設定ファイル等を用いる。測定プログラムは Firefox の動作不全を発見すると、Firefox のプロセスを終了し、正常に動作する Firefox の設定ファイルを上書きし、Firefox を再起動する。これにより、Firefox を再び正しく動作させることができる。また、FTP サーバとのコネクションが切断された場合も、再接続を行う。

3.2 収集条件配信サーバ

収集条件配信サーバは、測定プログラムに対して測定条件を配信する Web サーバである。測定条件には測定対象の URL と測定開始時刻等が含まれる。測定条件の例を図 4 に示す。測定条件は JSON 形式で記述さる。例では 2 行目に測定ホスト内で測定データが保存されるディレクトリのパス、3 行目に Firefox を起動するためのパス、4 行目に測定ホストの緯度経度を配信する URL、5 行目に Firefox が正しく動作するコンフィグファイル等のパスが記述される。また、6~10 行目には測定データ収集・解析サーバに測定データをアップロードするための FTP サーバについての情報が記述される。加えて、11 行目以降で複数の測定スケジュールを設定できる。測定スケジュールは測定開始時刻、測定時のタイムアウト時間、測定対象の URL リストを配信する URL で構成される。これらにより、測定プログラムは指定された現地時刻で、対象 URL の Web パフォーマンスを測定できる。また、Firefox の起動・再起動、測定データのアップロードが可能となる。

3.3 Web パフォーマンス測定プラットフォームと要求条件との対応

以上のような Web パフォーマンス測定プラットフォームは、先に挙げた要求条件を満足するものとなる。まず、測定データについては、Firefox を用いた HAR の収集と、各 Web サーバに対する RTT の測定により、要求条件を満足している。次に、

Web ブラウザの動作環境については、PlanetLab を用いることで、多様な位置、多様な処理性能のホストを用いることができ、動作環境の多様性を確保することができる。また、測定時刻に関しても、測定条件配信サーバにスケジュールを設定することで、各地点の現地時刻で測定を開始でき、少ない稼働で多様な時刻で測定可能である。加えて、Firefox や FTP の不調時には自動復旧されるため、稼働をかけることなく、測定プラットフォームの可用性を高めている。

4. 評価

提案プラットフォームの有効性を評価するため、実際に公開されている Web ページの Web パフォーマンスを測定することで、収集にかかる稼働、Web ブラウザ動作環境の多様性、可用性を確認した。加えて、収集された測定結果を解析し、Web パフォーマンスの改善方針の検討できることを確認した。

4.1 測定にかかる稼働の評価

測定を開始するには以下の作業が必要である(カッコ内は作業時間の目安)。

- (1) 測定プログラムを各測定ホストにアップロード (5分程度)
- (2) 測定条件を作成 (条件の内容に依存)
- (3) 測定ホストの cron を設定 (10分程度)

(1), (3) については、pssh を用いることで複数の測定ホストに対して同時にコマンド実行し、作業にかかる時間を短縮している。また、測定条件については、図4のファイルと、URL を羅列したファイルを作成し、収集条件配信サーバにアップロードすればよい。なお、1回の測定のためのこれらの設定にかかる時間は、測定条件にもよるが、概ね30分程度である。

4.2 可用性の評価

可用性を評価するため、長時間の測定の成功率と、測定中の自動復旧の回数を測定した。長時間の測定のために、多くの URL を測定対象とする必要がある。また、より実際の測定に近づくため、実際にサービスを提供しており、多くの人が利用する様々な Web ページを対象とする必要がある。このため、Alexa [7] のランキング上位の Web ページの URL を測定対象とした。なお、様々な Web ページを対象とするために、Web ページのカテゴリ (ニュースやショッピングなど) ランキング上位から同数ずつ抽出し、959 個の URL リストを作成した。

以上の測定を 108 の測定ホストにおいて実施した。この結果、測定が完了したホスト数は 101 であった。なお、測定を完了しなかった 7 ホストのうち、6 ホストはタイムゾーン取得の失敗や環境構築の誤りなどにより、測定を開始できていなかった。つまり、測定開始後に測定を完了できなかったホストは 102 ホスト中 1 ホストのみであった。測定が完了できなかった 1 ホストについては、動作が不安定なホストであり、何らかの理由で測定プログラムが停止したと考えられる。

4.3 測定成功率

959 個の URL のうち、測定に成功した個数について、平均値：544、中央値：559、最小値：319、最大値：635 という結果となり、多くの URL で測定に失敗していることがわかる。

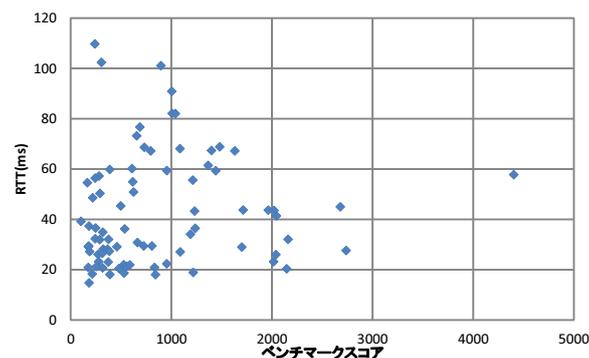


図5 測定ホストの RTT とベンチマークスコアの分布

HAR の収集や RTT の測定がタイムアウト時間満了までに完了しないことが多く、収集に失敗していると考えられる。なお、復旧回数の中央値は 400 程度であり、このような収集の失敗のたびに復旧が行われていると考えられる。

4.4 測定時間

959 個の URL の測定にかかった時間 (各 URL の測定のタイムアウト時間は 30 秒) について、平均値：13:00、中央値：12:38、最小値：9:17、最大値：19:38 という結果となった。この結果から、1URL の測定には 50 秒程度必要であることがわかる。これは、HAR の収集に数秒~30 秒程度、RTT の測定等の収集に数秒~数十秒、FTP による測定結果のアップロードに数秒程度かかることから妥当であると考えられる。

4.5 Web ブラウザの動作環境の多様性

提案プラットフォームは、ネットワーク内での端末の位置、測定開始時刻、端末の処理性能といった条件の組み合わせにより、多様な Web ブラウザの動作環境を提供する。図5に測定ホストごとの RTT とベンチマークスコアの分布を示す。RTT は先の 959 個の URL の Web ページを表示する際にオブジェクトをダウンロードする全ての Web サーバに対する RTT の平均である。図5を参照すると、RTT、ベンチマークスコア共に広く分散しており、多様な Web ブラウザの動作環境を提供できていることがわかる。なお、測定時刻については測定条件に記述される測定開始時刻に多様な時刻を記述することができる。

4.6 測定データの解析結果の例

以上の測定において収集されたデータの解析結果の例として、Web ブラウザの動作環境と Web パフォーマンスとの関係を確認した。表1に onload 時間と RTT・ベンチマークスコアとの間の相関係数についてまとめた。なお、表の+は onload 時間と正の相関がある (相関係数が 0.2 以上) ことを示し、-は onload 時間と負の相関がある (相関係数が -0.2 以下) ことを示す。また、空欄は無相関であることを示す。表の各値はそれぞれの相関係数を持つ Web ページの数を表し、例えば onload 時間と RTT とは負の相関があり、onload 時間とベンチマークスコアとも負の相関がある Web ページは 83 個あることがわかる。

このような onload 時間と RTT・ベンチマークスコアとの相関係数を算出することで、RTT と端末の性能とのどちらが Web パフォーマンス (onload 時間) に影響を与えるかを特定することができる。これにより、ある程度の Web パフォーマンス

スの改善方針の検討が可能となる。例えば、onload 時間とベンチマークスコアとに負の相関があれば、端末での Web ページのレンダリング処理に時間がかかっており、Web ページの構造を改善することとなる。また、onload 時間と RTT とに正の相関があればオブジェクトのリクエストや転送に時間がかかっていると考えられ、CDN の利用などを検討することとなる。一方、RTT と負の相関がある場合、RTT が長くなることでダウンロードするオブジェクト数が少なくなり、onload 時間が短くなることが考えられる。具体的には、RTT が長いと広告が表示されないことがあり、表示される場合と比較して onload 時間が短くなる。

以上のように、測定データの解析結果の例を示したが、相関係数の算出対象の値をサーバでの処理時間 (Wait 時間から RTT を差し引いた値) や各内訳の時間などにすることで、別の観点での解析が可能となる。また、相関係数による解析以外の要因分析に収集データを利用することも考えられる。

5. 関連研究

Web パフォーマンスの測定と、ボトルネックの発見については様々な研究が行われてきた。

Meenan は Web パフォーマンスのアクティブテスト手法を体系的に紹介している [8]。また、実践的な Web パフォーマンス向上手法がまとめられた参考書として、Grigorik による High Performance Browser Networking [1] や Souders による High Performance Web Sites [2] が有名である。

Web パフォーマンスを測定するツールも登場しており、YSlow [9] や Chrome DevTools [10] は広く利用されているほか、自身が提供する Web サイトが実際にユーザのブラウザで表示される際のパフォーマンスを測定するための API として Navigation Timing [11] や Dhawan らは Fathom などがあり、Web パフォーマンスの測定環境は充実してきている。Fathom で Web ページを実装することで、その Web ページを表示した際に、Web ブラウザは各種測定結果をサーバに送信する。これにより、Web ページの提供者は、実際にページが表示される際のパフォーマンスを知ることができる。加えて、Web パフォーマンスのボトルネックの診断ツールも開発されている。Wang らは Web ページを構成するオブジェクト間の依存関係の種類を分類・定義し、それに基づき Web ページを構成するオブジェクト間の依存グラフを生成することで Web パフォーマンスのボトルネックを発見するプロファイラである WProf を開発している [3]。

様々な場所から Web パフォーマンスを測定する環境についても研究されている。Sundaresan らは各家庭のブロードバン

ドルータに測定プログラムをインストールすることで、有名な 9 個の Web ページのパフォーマンスを測定し、ボトルネックを特定した [12]。

以上のように、様々な Web パフォーマンス測定、ボトルネック発見手法について研究されているが、様々な Web ブラウザ動作環境で、特定の API で実装されていない任意の Web ページのパフォーマンスを測定するための手法は提案されていない。

6. おわりに

本稿では、Web ブラウザの動作環境を変化させ、且つ、Web パフォーマンスに加えてサーバ・ネットワーク・クライアントそれぞれのパフォーマンスについても測定可能とする、Web パフォーマンス測定プラットフォームを提案した。提案手法を測定にかかる稼働、可用性、測定成功率、測定時間、Web ブラウザ動作環境の多様性、測定データの解析結果の観点から評価した。この結果、提案手法は 1 回の測定にかかる稼働は 30 分程度と少なく、959 個の Web ページを 102 のホストで測定した結果、101 のホストで測定を最後まで継続できることを確認した。また、PlanetLab を用いることで、多様な Web ブラウザ動作環境での測定を実現した。これにより、測定結果から、Web ページのパフォーマンス改善の方針を検討できるデータを収集することができた。一方、959 個の Web ページのうち、400 程度の Web ページで測定がタイムアウトすることで、測定に失敗することがわかった。測定成功率向上のため、今後の改善が必要である。

文 献

- [1] I. Grigorik, High Performance Browser Networking What every web developer should know about networking and web performance, O'Reilly Media, 2013.
- [2] S. Souders, High Performance Web Sites Essential Knowledge for Front-End Engineers, O'Reilly Media, 2007.
- [3] X.S. Wang, A. Balasubramanian, A. Krishnamurthy, and D. Wetherall, "Demystifying page load performance with wprof," Presented as part of the 10th USENIX Symposium on Networked Systems Design and Implementation (NSDI 13), pp.473–485, USENIX, Lombard, IL, 2013.
- [4] "Http archive (har) format". <https://dvcs.w3.org/hg/webperf/raw-file/tip/specs/HAR/Overview.html>
- [5] "Planetlab". <https://www.planet-lab.org/>
- [6] "Geonames". <http://www.geonames.org/>
- [7] "Alexa". <http://www.alexa.com/>
- [8] P. Meenan, "How fast is your website?," Commun. ACM, vol.56, no.4, pp.49–55, April 2013.
- [9] "Yslow". <http://yslow.org>
- [10] "Chrome devtool". <https://developer.chrome.com/devtools>
- [11] "Navigation timing 2". <http://www.w3.org/TR/navigation-timing-2/>
- [12] M. Dhawan, J. Samuel, R. Teixeira, C. Kreibich, M. Allman, N. Weaver, and V. Paxson, "Fathom: A browser-based network measurement platform," Proceedings of the 2012 ACM Conference on Internet Measurement Conference, pp.73–86, IMC '12, ACM, New York, NY, USA, 2012.

表 1 onload 時間と RTT, ベンチマークスコアとの間の相関の有無 (相関係数の絶対値が 0.2 以上で相関有りと判定)

| ベンチマークスコア \ RTT | + | - | |
|-----------------|----|-----|----|
| + | 1 | 0 | 1 |
| | 6 | 10 | 3 |
| - | 10 | 147 | 83 |