

大規模イベントにおける準リアルタイム渋滞情報提供システムの 構築と運用

寺元 貴幸* 福谷 遼祐**

Construction and Operation of Near-real-time Congestion Information Providing System in Large-scale Events

Takayuki TERAMOTO* and Ryosuke FUKUTANI**

In recent years, it is very major method that use internet for collecting information. Especially, social network service (SNS) is growing rapidly. For example, there are Twitter, Facebook, LINE, mixi, and so on. However, SNS is very difficult to understand for Information illiterate. In this challenge, in order to provide real-time information of "B-1 Grand prix in Tsuyama", our system was built. This system provides near-real-time information. Information is based on information that was sent by staff. Moreover, user interface is simple web-page and it is easy to use. In order to get latest information, the web browser is only required. In the day of event, our system acted role without incident. It seems without problem that operating our system, but exists some issues about construction. In this report, our system, operation, and consideration were recorded.

Key words: Web, Real-time, Provide-Information, Smart-phone

1. 概 要

近年, Web は情報提供の手段として定着してきており, そのなかでも Twitter や Facebook, LINE などに代表される SNS の普及によるユーザ同士の情報伝達も一般的になってきた. しかし, これらのサービスを活用するためには, 事前にサービスの利用登録を行うことが基本的に必要となり, 正確でしかも常に情報が更新されるタイプの情報提供は期待することができない. さらに, 短期間のイベントでは, 常設の駐車場のよう渋滞情報を自動計測するようなシステムを導入することは困難である. そこで今回, 大規模なイベントにおいても, ほぼリアルタイムに駐車場の渋滞情報や店舗の混み具合を知る事ができ, しかもレンタルサーバとスマートフォンを用意するだけで, 非常に安価に運用できるシステムを開発したので報告する.

今回本システムを 2013 年 5 月 25・26 日に実施された「近畿・中国・四国 B-1 グランプリ in 津山」にお

いて運用することを試みた. B-1 グランプリにおいては, 以前から Web ページによる情報の提供自体は行われていたが, スタッフ同士がメールを用いて情報交換を行い, これに従ってスタティックな Web ページを逐次書き換えて表示する情報の更新を行っていた. しかし, この方法では手間がかかり, リアルタイム性も非常に低くなりがちであった. そこで今回本システムを適用することにより「スタッフが現在の情報を送信すると Web ページを最新の情報に更新」というサイクルを自動化するためのシステムによりリアルタイム性の高い情報提供を行うことができたのでその様子を報告したい.

2. システム構成

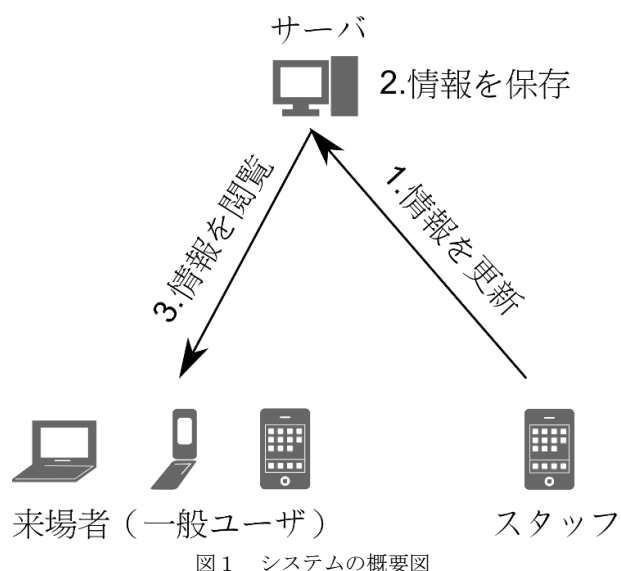
2. 1. システム全体と運用方法

本システムでは, Web サーバが情報を直接受け取り, 即時に Web ページを書き換えるためにスクリプト (PHP) を用いた動的な Web ページによる情報の提供を行う. 以下の図 1 に概要図を示す.

原稿受付 平成 25 年 8 月 30 日

*情報工学科

**電子・情報システム工学専攻科



また、実際の運用の手順は以下のとおりである。
 (1) スタッフが、スタッフ専用ページにスマートフォンなどを用いてアクセスし、任意の情報を最新のものに更新する。実際のスマートフォンでの操作画面を以下の図2に示す。

図2 情報更新画面

(2) Web サーバは、情報を最新のものに更新し、以前の状況をバックアップする。

(3) 一般の利用者は、来場者用のアドレスにアクセスし、最新の情報を取得する。実際の画面以下の図3に示す。

図3 情報閲覧画面

今回、開発を行ったシステムは基本的に「情報の更新と取得を行うライブラリ」、「スタッフが情報を更新するためのページ」、「一般の来場者が情報を閲覧するためのページ」に分けることができる。また、ライブラリを含むシステム群は今回の B-1 グランプリに限らず、ほかの大規模イベントにも対応できるように、提供すべき情報の数が増加した場合でも容易に拡張が可能となっている。以下の表1に今回のイベントで提供した情報を示す。

表1 提供した情報

種類	個数
駐車場の混雑状況	3
シャトルバスの待ち時間	2
チケット売り場の待ち時間	4
出展団体の待ち時間	22

以上のように、合計 31 箇所の情報についてリアルタイム性の高い情報の公開サービスを提供した。

2. 2. 情報の更新と取得を行うライブラリ

Web サーバ上の情報を更新したり、最新の情報を取得したりするために各 Web ページ上で使用される共通の関数群がまとめられた PHP ファイルである。Web サーバは、各種の情報を保持しておく必要があり、TXT ファイルで現在の情報を保存することとした。このファイルには、フォームで情報を更新する際の一意の ID と最新の情報、更新時刻などが含まれている。情報公開用サイト内で使用する、逐次更新されるべ

き情報は全てこのファイル内で一括管理されている。また、性能向上のため APC (Alternative PHP Cache) と呼ばれる PHP を高速化するモジュールを有効化した。これに含まれる、`apc_store` という関数によって TXT ファイルの内容を、セッションをまたいでメモリ上に保持することが可能となった。このため、一般ユーザからのアクセス時には TXT ファイルを読み出す必要がなくなり、TXT ファイルの読み書きは情報の更新時のみでよかった。

2. 3. スタッフが情報を更新するページ

スタッフが最新の情報を任意のタイミングで Web ページに反映させるために、情報を Web サーバに送信するためのページである。一般のユーザが利用すべきページではないため、HTTP リクエストの GET パラメータを用いて簡易的なパスワードを設けている。このため、仮にスタッフ専用の URL を推測できたとしても、GET パラメータにセットすべき値が漏洩しない限り、スタッフ専用のページにアクセスすることはできない。また、GET パラメータの値を正しく設定したとしても、セッションごとにパスワードによる認証が必要である。このパスワードは、実際の運用時はスタッフの手間を軽減する目的で無効化されていた。

また、今回のシステムでは、上記のようにスタッフの手間を軽減することを主な目的のひとつとしてある。このため大勢存在するスタッフは自身の担当すべき情報以外の情報も任意に更新することが出来る。これはセキュリティ上好ましくなく、本来ならスタッフは個別に ID とパスワードを用いて認証を行うべきであるが、今回のケースでは実装の手間と利用の手間がどちらも増加することになるため、特に個別の認証は実装されなかった。

2. 4. 一般の来場者が情報を閲覧するページ

一般の来場者が使用する情報公開用ページである。先に述べたライブラリにより、ページ上の情報はアクセス時に最新のものが適用される。単純に PHP の `echo` コマンドを使用しているため、ユーザから見るとスタティックなページと変わりはない。

2. 5. その他

(1) エラーページのカスタマイズ

Apache 標準のエラーページではなく、利用者がわかりやすいデザインのページが表示されるように変更した。

(2) PC, スマートフォン, フィーチャーフォンそれぞれに最適化されたページの表示

PHP のユーザーエージェント判別機能により、それぞれに適したページが表示されるようになっている。

3. 実際の運用

本システムを運用するため、専用レンタルサーバを Web サーバとして稼働させた。ここで、一般的なレンタルサーバでは APC が使用できなくなり、性能が低下することに注意すべきである。以下の表 2 に APC が有効化された場合と無効された場合での性能を示す。この時の性能とは、PHP によって動的に生成されるページを大量にリクエスト(合計 10000 リクエスト, 同時接続数 20)を行った場合の秒間アクセス数とする。この検証には Web サーバなどの負荷テストツールの一つである `ab` を用い、さくらインターネットの VPS1G プランでの性能を比較した。

表 2 性能の比較

	秒間アクセス数 (エラー含む)	秒間アクセス数 (エラー除く)
APC なし	47 毎秒	6 毎秒
APC あり	54 毎秒	54 毎秒

以上のように、APC によって大幅な高速化を行うことができたことがわかる。また、負荷テストに使用したネットワーク帯域の使用率がほぼ 100%に達していたため、実使用環境ではさらに性能が伸びる可能性がある。以上のテストは VPS を用いたため、使用できる帯域が不安定であるが、実際に運用に使用した専用サーバ上では、仮に 100Mbps の転送速度を前提にした場合、約 2800 毎秒ものリクエストを処理できる性能が発揮できる見込みとなった。また、実際に使用したサーバの仕様を以下の表 3 に示す。

表 3 使用したサーバ

サービス名	さくらインターネット さくらのマネージドサーバ Xeon プラン
使用料金	月額¥14800 初期費用¥30000
root 権限	なし
マシンスペック	CPU:Xeon, RAM:8GB, HDD:600GB
ネットワーク	100Mbps 共用

イベント当日の運用については、特に大きな混乱もなく行われた。以下の図 2 にイベント時のサイトへのアクセス数の推移を示す。今回のイベントでは、日中において秒間 1~2 程度のアクセスしかなかったため、サーバに対しての負荷はほとんどかかっていなかった(アイドル時と同程度)。このため、使用したサーバの性能は十分であったと思われる。

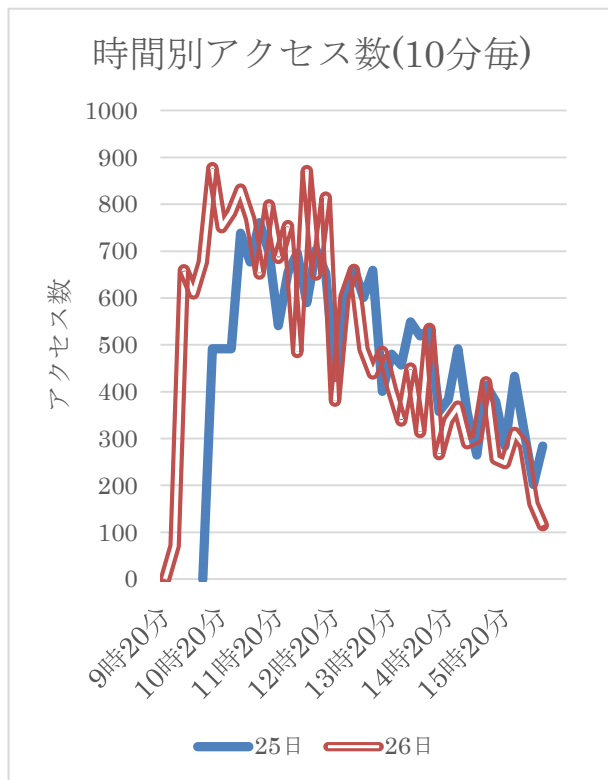


図2 時間別アクセス数

6. ま と め

今回の例では、もともと人が行っていた作業を Web サーバ上で自動化することが主な目的であった。PHP によるサーバサイドの処理によって自動化は実現できたものの、以下のような課題も明らかとなった。

まず、セキュリティ上の脆弱性に対しての問題があげられる。情報を更新するスタッフに対して認証を行わないため、だれが何を更新したのかわからない。パスワードの入力などの煩雑さも、Web ブラウザによるパスワードの保存機能などを用いることで低減できると考えられる。

次に APC 実行可能環境に依存しなければならない問題がある。今回のシステムでは、大量のアクセスに備えて APC を導入したが、果たして十分に効果があったかどうか疑問である。それよりも APC を有効化できる環境が限られる不便さのほうが問題となりかねない。アクセス数の増加に対しては、それ以前

にネットワーク帯域の確保、サイトの効率的な設計などのほうが重視されるべきであった。しかしながら、特に大規模なイベントなどに対応するためにはやはり APC などの PHP アクセラレータが必要になってくる場合がある。この場合は、自分でサーバを構築するか、または一般的なレンタルサーバではなく VPS や今回利用したような専用サーバ(ネットワーク帯域が保障されていればなお望ましい)を使用して運用を行うべきである。もちろん、APC なしでも問題なさそうであれば無理に高価な環境を整える必要はないと思われる。

最後に、更新情報の管理の方法についての問題がある。今回のシステムでは、サイト上で一意の ID を持つ情報を単一の TXT ファイルに保存する手法をとった。しかしこれでは情報そのものの管理が煩雑になるため、各情報は階層構造によって管理されるほうが効率的であると考えられる。

今回のシステムにおいて、最も重要であるのは、情報の更新と取得が可能な PHP のライブラリを作成したことである。しかしこれは汎用性が低いわけではないが上記のような問題を抱えており、今すぐにほかのイベントなどで使用できるものではない。よって今後、汎用性を重視してシステムの設計・実装を変更していくべきである。

参 考 文 献

- 1) B-1 グランプリ公式サイト, <http://b-1tsuyama.com>
- 2) 清野克行：作りながら基礎から学ぶ PHP による Web アプリケーション入門, (2011)
- 3) 掌田津耶乃：HTML5 Web アプリケーションプログラミング入門, (2012)
- 4) 園田誠：HTML と CSS で基礎から学ぶ JavaScript, (2011)
- 5) PHP マニュアル, <http://php.net/manual>
- 6) さくらインターネット, <http://www.sakura.ad.jp>
- 7) Web クリエイターネット, <http://www.webcreator-net.com>
- 8) PHP リファレンス, <http://www.php-ref.com>
- 9) sakura vps manual, <http://www.sakura.vps.com>
- 10) PHP の拡張モジュール APC をインストール
<http://d.hatena.ne.jp/replication/20110327/130121832>