

PutItOnWeb を用いた情報送信システムの構築

Developing Information Sending System by PutItOnWeb System.

中京大学大学院情報科学研究科

木村訓康, 田村浩一郎

1. はじめに

Web を使ってグループが待ち合わせを行う例を考える。電子メール等 (FAX などを含む) を使い、地図の URL と説明文を書いてこれを送るという方法や、Web ページ上の地図に場所を示すマーカーをつけ、説明文を加え、その Web ページのアドレスを送るという方法がある。このとき、メンバーが受け取った待ち合わせの情報を元に、さらに情報を追加する (例えば質問や出欠の返事など) には再び同様の操作をしなければならない。

そこで、最初は Web ページ上に情報を付加する機能をもったシステムを用意し、このシステムで情報付加した Web ページのアドレスを電子メール等によってメンバーに渡し、その後はこのシステムにより情報を付加する方法により、グループでの情報共有を円滑に行えるシステムを考えた。

2. Web ページへの情報付加

PutItOnWeb システム[1]は、任意の Web ページに PutIt と呼ぶ仮想付箋紙を貼り付けることで

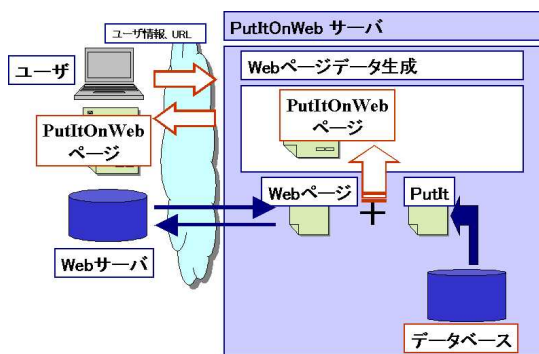


図1 PutItOnWeb システム図

Web ページに情報を付加し、ユーザ同士のコミュニケーションを図るためのシステムである。ユーザはユーザ情報 (ユーザ ID とパスワード) と Web ページの URL を入力し PutItOnWeb ページ (PWP) を Web ページとして取得する。サーバは、ユーザが入力した URL の Web ページを取得し、URL とユーザ情報をキーに PutIt のデータをデータベースから取得し、この2つにクライアントサイドプログラムをあわせて PWP を作り、ユーザに送信する (図1)。

3. PWP のアドレスと属性

グループのメンバーは PWP の識別子を付加したアドレスを用い PutItOnWeb サーバにアクセスする。サーバはこの識別子から PWP に含むべき PutIt と URL をデータベースから取得し PWP を生成する。この PWP がグループ間での情報共有を行う対象となる。

PWP はあて先を示す属性値として、送信するメールアドレスと、閲覧を許可するユーザを指定する。前者はサーバにより PWP アドレスを自動的にメールとして送信するとき用い、後者は PWP へのアクセスを厳密化するために用いる。

4. クライアントサイドプログラム

PutItOnWeb システムでは、PWP に含まれたクライアントサイドプログラムによって、ビューに対する PutIt の貼り付けや PutIt への各種操作 (PutIt のドラッグや新規作成等、仮想付箋紙としての機能を与える) が行われる。さらに PWP の属

性を設定する機能をこのプログラムに持たせ、ユーザが任意に設定できるようにした。具体的には、送信したい PutIt を右クリックで選択し、フォーム入力によって先の入力を行う。クライアントサイドプログラム中でこれらの値を変数として持ち、ユーザの指定したタイミングで POST を使ってサーバに送る。

5. 動きのある Web ページへの導入

PutIt は、GoogleMapsAPI[2]のような AJAX 等を用いた動きのある Web ページの DOM オブジェクトに貼ることもできる[3]。PWP では DOM オブジェクトの初期表示範囲に PutIt が貼られているとは限らない。これでは、メール等を使いアドレスが送られてきても PutIt を見ることができず、情報共有を行うことができない。GoogleMapsAPI を利用したページに限定されるが、PutIt を表示することができる範囲に DOM オブジェクトの表示範囲を調整するプログラムを PWP 組み込みのクライアントサイドプログラムに追加した。このプログラムは表示範囲を、与えた緯度、経度を中心とするものに変更する。ここで、緯度、経度として PutIt の座標を与えることで少なくともひとつの PutIt を表示することができる。この値を PWP の属性値のひとつとする。

また、この機能を使い PWP の表示で中心となる PutIt を変えることができる。たとえば PutIt A に PutIt B の識別子を持たせることで PutIt A をダブルクリックしたとき表示の中心を PutIt B にする。

6. 実装と検証

PutItOnWeb サーバシステムの実装に Ruby on Rails[4]を用いた。また識別子に UUIDTools[5]による UUID を用いた。データベースとして MySQL 5.0.26 と Oracle9i の両方が使用できる。検証には MySQL5.0.26 を用いた。PWP 組み込みのクライ

アントサイドプログラムは JavaScript とそのライブラリとして Prototype.js v1.5.0 を利用した。サーバの CPU は Pentium4 1.6Ghz でメモリ 512M である。公開用に Mongrel を用いた。

この環境で実装を行ったシステムに対して、Firefox2.0.0.3 を使い数十人のユーザに利用してもらい検証を行った。データベースに 1000 件のダミー PutIt をもち、その状態で PWP のアドレスを入力してから PWP が作られるまでの平均時間と、サーバに要求がきてから PWP のアドレスを送信するメールが送られるまでの平均時間を測定した。PWP が作られる時間の平均時間は PWP に含まれる PutIt が 1 個のとき 0.692sec、10 個のとき 0.738sec であった。メールが送られるまでの平均時間は PutIt が 1 個のときが 0.772sec、PutIt が 10 個のときが 0.813sec であった。PutItOnWeb システムでユーザが入力した URL をサーバが受け取り PWP を作るまでの時間が 0.638 であったことを考えても十分な数字と言える。また、どの場合も最初の一人のアクセスに時間がかかる。二人目以降はキャッシュを使うことで、データベースからデータを取得する時間や、Web ページを取得してくまでの時間が短縮できるためと考えられる。

参考文献

- [1] 木村,田村:「Put It On Web - Web ページに情報を付加することによる情報共有の手法 - 」, 第 58 回グループウェアとネットワークサービス他合同研究発表会, Jan, 2006
- [2] <http://www.google.com/apis/maps/>
- [3] 後藤,田村,木村:「Put It On Maps」,FIT2007 発表予定, Sep, 2007.
- [4] <http://www.rubyonrails.org/>
- [5] <http://sporkmonger.com/projects/uuidtools/>