

Web ページ再活性化システムを利用した Web ページ上での 協調ワークスペース

湯浅 康史[†] 土井 達也^{††} 白松 俊^{††} 大園 忠親^{††} 新谷 虎松^{††}
名古屋工業大学工学部情報工学科[†] 名古屋工業大学大学院工学研究科情報工学専攻^{††}

1 はじめに

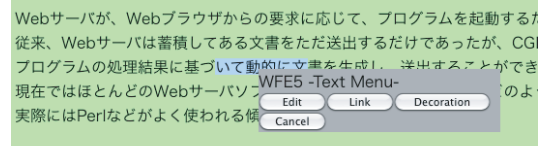
CMS を導入していない Web ページの編集は Web ページの編集の知識が無い人には難しく、多くの Web ページが更新されずに放置されている問題がある。そこで本研究では、既存の Web ページをブラウザ上から編集可能にすることで Web ページを簡単に編集可能にする WFE[1] を元に、より簡単に導入可能な Web ページ再活性化システム [2] を開発した。Web ページ再活性化システムを導入した Web ページでは、Web ページをブラウザ上から編集可能になるだけでなく、Web ページの編集情報を同じ Web ページを閲覧しているユーザに自動的に同期する。Web ページをブラウザ上で編集可能になる、編集情報を他の閲覧者に自動的に同期するという特徴から、本システムを利用することで既存の Web ページを簡単に協調編集可能な Web ページにすることが可能となる。

2 Web ページ再活性化システム

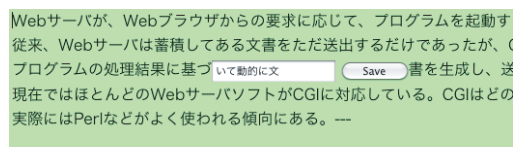
Web ページ再活性化システムは Web ページを Web ブラウザ上から編集可能にするシステムである。ブラウザ上で Web ページを編集するシステムには Wiki, blog, SNS 等がある。これらのシステムは、そのシステム内のコンテンツを編集するが、本システムでは既存の HTML で書かれた Web ページを編集可能にする。本システムを導入した Web ページでは、閲覧時にブラウザ上で書き換えたい部分を選択して右クリックすることでメニューウィンドウ (図 1.(a)) が開く。メニューウィンドウから Edit を選択することで編集ウィンドウ (図 1.(b)) が表示される。編集ウィンドウに書き換えたい内容を書き込むことで、容易に Web ページを編集可能にする事が出来る。

WFE[1] では Web サーバに CGI を設置する必要があった。CGI を動作させるには、動作環境を整備する必要があり、レンタルサーバ等においては CGI が動作しない可能性がある。しかし Web ページ再活性化システムではコンテンツ管理サーバが CGI の処理を行うため、Web サーバに CGI 等のファイルを設置する必要がない。Web ページにこのシステムを導入するには、Web ページのソースコードに再活性化のために必要な JavaScript をコンテンツ管理サーバからダウンロードするためのタグを記述するだけでよい。そのため Web サーバにいくつかのファイルを設置したり、CGI 環境を整備する必要が無く簡単に導入することが可能である。

WFE ではクライアントが編集した情報を Web サーバ上の HTML ファイルに同期するために、Web ブラウザ上で動作している JavaScript が編集情報を Web サーバに送信し、Web サーバの CGI が受け取った編集情報から HTML 情報を書き換える。しかし Web ページ再活性化システムでは HTML ファイルと CGI は異なるサーバ上に配置されている。そのため、CGI は HTML ファイルを書き換えることが出来ない。



(a) メニュー画面



(b) 編集画面

図 1: Web ページ再活性化システムの編集画面

そこでコンテンツ管理サーバに Web ページのソースコードを保存するデータベースを設置している。Web ページのかわりにデータベースに保存されている Web ページの情報を利用する。

3 既存 Web ページの協調ワークスペース化

Web ページ再活性化システムを導入した Web ページでは、Web ページが編集されたとき、他の閲覧者にその編集情報が同期される。また編集者の制限を設けなければ複数のユーザが Web ページを編集可能になる。これらの特徴を利用することにより、既存の Web ページを複数のユーザが協調してファイルを編集する協調ワークスペースとして利用することが出来る。例えば Web ページを作るとき、複数のユーザで同時に編集しながら協力して一つの Web ページを作ることが可能となる。Web ページを会議資料として利用する場合、発表者が Web ページを資料として利用するだけでなく、視聴者が資料をブラウザで閲覧できる。さらにこのシステムを導入することにより、閲覧者が伝えたいことを Web ページに書き込んだり、会議の決定事項を資料を書き換える事ができる。

このような多人数での Web ページの編集を補助する機能としてコメント機能がある。コメント機能では Web ページの好きな位置に付箋のようにコメントを付けることが出来る。コメント機能を利用することで閲覧者同士でのコミュニケーションを取ることが出来る。

4 協調ワークスペースにおける同期

本システムにおいて、編集情報が同期されるまでの時間が長いと、衝突発生期間が長くなり衝突確率が上昇する、反映の遅さからユーザビリティが低下するといった問題がある。同期速度を向上させるために Web ページ再活性化システムに Comet を導入した。Comet とはプル型通信である HTTP において擬似的にプッシュ型通信を実現する技術である。あ

Cooperative Workspace on a System for Reactivation of Web Page
Yasufumi YUASA, Tatsuya DOI, Shun SHIRAMATSU, Tadachika OZONO, and Toramatsu SHINTANI

Nagoya Institute of Technology, Dept. of Computer Science and Engineering, Dept. of Computer Science and Engineering, Graduate School of Enngearing, Nagoya Institute of Technology.

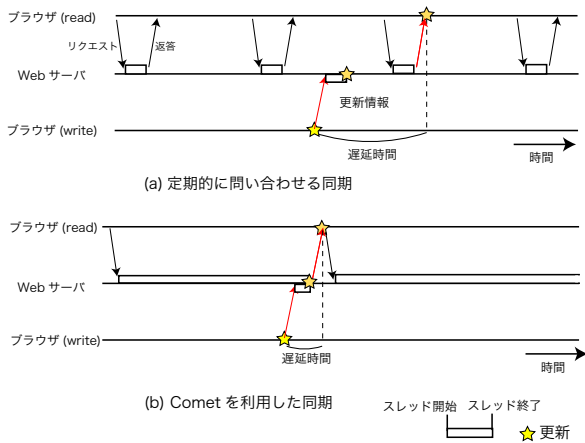


図 2: プッシュ型通信による同期速度

あらかじめクライアントがリクエストを出す。クライアントのリクエストに対してデータを送りたいタイミングまで返さずに保留する。タイムアウトになるまでの間、サーバはいつでもクライアントに送りたい情報を送ることが出来る。この流れによりプル型通信における HTTP においてプッシュ型通信を擬似的に実現できる。プル型通信における同期では、定期的に Web サーバに更新情報があるか問い合わせる手法になる。この手法では、Web サーバ上のデータが更新されてもクライアントが問い合わせるまでサーバは編集情報を送信できずに、同期の遅延が発生する(図 2.a)。さらに更新が無いときにも問い合わせるため無駄な通信が発生する。Comet における同期では、クライアントの編集情報の問い合わせに対して、Web サーバはすぐには返答しない。Web サーバのデータに更新が起きたとき、サーバは保留していたクライアントのリクエストに対して編集情報をすぐに返す。編集と同時に返答するため、問い合わせを待つ必要がなく遅延は減少する(図 2.b)。また無駄な問い合わせも無く、通信量を抑えることが出来る。

複数のユーザが同じ Web ページを同時に編集するにあたって、ユーザの編集情報が上書きされてしまい編集情報が失われる編集の衝突の問題がある。Web ページ再活性化システムでは文献 [2] で触れているように、DOM 要素単位の編集情報を送信している。DOM 要素単位で編集情報をやりとりすることにより、同時に編集した場合でも Web の異なる部分を編集していた場合、衝突は発生せず衝突確率を減らすことが出来る。

5 評価・考察

本システムでは複数のページの編集処理を一つのコンテンツ管理サーバが担う。そのためコンテンツ管理サーバにアクセスが集中することが考えられる。また、Comet では、リクエストを受けて呼び出された CGI プロセスは更新があるまで返答せず終了しないので、Web サーバでは一つの閲覧に対して一つのプロセスが起動していることになる。よって特徴からコンテンツ管理サーバに負荷がかかりレスポンスが遅くなる可能性がある。そこで接続数を増加させた時の、コンテンツ管理サーバの CGI にデータベースに格納されているファイルを要求し帰ってくるまでの時間、および CPU 使用率の増加を計測した。サーバとして CPU に 2.66Ghz Dual-Core Intel Xeon を二つ、2GB のメモリ、Mac OS 10.6.5 を搭載している PC を利用し実験を行った。実験結果を図 3 に示す。図 3.a では 30 人程度までの接続数ではレスポンスタイムの増加は緩やかであったが、30 人以降は接続数の増加に対するレスポンスタイムの増加が少し急になっている。CPU 使用

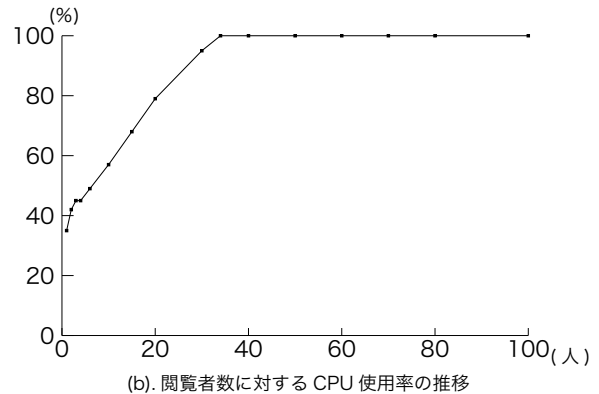
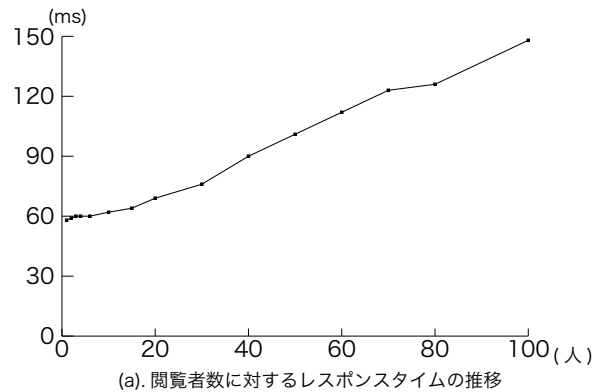


図 3: 閲覧者数を増加させたときの負荷の推移

率は 35 人付近で 100% になっており、30 人以降のレスポンスの悪化は、CPU 使用率が主な原因である事が考えられる。この結果からこのシステムは 35 人程度までの接続数に影響なく耐える事が可能であると考えられる。プロセス毎の CPU 使用率を調べたところ、接続数が CPU 使用率が 100% になる 35 人のときデータベースが約 52%、CGI が約 14% であった。接続数が 100 人のときデータベースが 62%、CGI が 11% であった。この結果からデータベースの処理が CPU の負荷の主な原因となっていることが分かる。データベースのアクセスを減らすように CGI のアルゴリズムを改良することにより全体の負荷が減少し、接続人数を増加させることが可能であると考えられる。

6 おわりに

本稿では、既存の Web ページを簡単に Web ブラウザ上で編集可能にする Web ページ再活性化システムを利用し、協調ワークスペースとして利用する方法を提案した。協調ワークスペースとして利用する際に課題となる同期の問題に触れ、同期速度を向上する手法を提案した。また本システムを利用する上で考えられる負荷についての評価、考察を行った。今後、接続可能人数を増加させるための改良が課題となる。

参考文献

- [1] 西 健太郎, 新谷 虎松, 松尾 徳朗, 田代 慎治, 伊藤 孝行: “既存 Web ブラウザを利用したオンライン編集可能な Web ページの実現”, 電気学会論文誌 (部門誌)C. vol.125, no.4, pp.660-665, Apr. 2005.
- [2] 土井 達也, 湯浅 康史, 白松 俊, 大冨 忠親, 新谷 虎松: “既存 Web ページの利用を活発にするための Web ページ再活性化システム”, 第 73 回情報処理学会全国大会, Mar, 2011 (掲載予定).