1Z-02

WebSocket を用いたリアルタイムな Web デスクトップ共有

鈴木　啓真†　　兼子　正勝†

電気通信大学情報理学研究科総合情報学専攻†

1　はじめに

近年，Web アプリケーションが注目されている．そしてそれを拡張し，コンピュータのデスクトップ画面そのものを Web ブラウザ上で作動させるという理念に基づいた「Web デスクトップ」と呼ばれるシステムが登場した．Web デスクトップの例として Netvibes[1]や AstraNOS[2]がある．しかし，これらのサービスはプライベート的な利用方法を想定している．そこで Web の特性を生かし，Web デスクトップに共有機能を追加した「共有型 Web デスクトップ“AppTop”」が鈴木ら[3]により開発された．共有型 Web デスクトップでは 1 つのデスクトップ画面を複数人で利用することができる．これによりグループ活動を支援するグループウェアとして利用する等，柔軟な利用方法が可能となる．しかし，AppTop では複数人が同時に編集を行うと，編集者以外のユーザにはリアルタイムに変更点が反映されず，反映のために各ユーザは手動でリロードしなければならないという問題がある．そのため，1 つのデスクトップを同時に複数人で利用できるシステムとしては十分とは言えない．

一方，Web におけるリアルタイム性は注目されている．要因としては，今までは困難であったブラウザとサーバ間でのリアルタイム双方向通信が WebSocket と呼ばれる技術の登場により可能になったためである．しかし，共有型 Web デスクトップに WebSocket を適用した例はまだない．

そこで本論文では WebSocket を用いてリアルタイムに変更点が反映される共有型 Web デスクトップを提案する．

2　WebSocket について

WebSocket ではブラウザとサーバ間でコネクションが確立されると，切断処理を行うまでコネクションが維持されるため，リアルタイムに双方向通信を行うことができる点が特徴である．

Real-time communication for Shared Web Desktop using WebSocket
Hirochika Suzuki†, Masakatsu Kaneko†
†Dept. of Informatics, Faculty of Informatics and Engineering, the University of Electro-Communications

これにより従来の技術では難しかった，サーバからブラウザへの任意のタイミングによるデータ送信が可能となった．モダンなブラウザであれば特別なプラグインなどを導入することなく利用可能である．本論文ではこの特徴に着目してリアルタイム更新の実装に WebSocket を採用した．ただし，Web の通信には通常，HTTP プロトコルが利用されるが，WebSocket では専用のプロトコルが利用されるため，Web サーバとは別途，新たに WebSocket サーバを設ける必要がある．

3　システム概要

共有型 Web デスクトップでは，1 つのデスクトップ画面を複数人で利用することができる．つまり，図 1 で示したように，ユーザ A とユーザ B のブラウザには同様のデスクトップ画面が表示されている．そして，本システムでは WebSocket を利用することで，このデスクトップ画面上で片方のユーザが変更を与えた場合に，その変更点をリアルタイムにもう片方のユーザに送信し，反映することができる．具体的には，アプリケーションの配置の変更，アプリケーションの追加・削除，背面のホワイトボードに描画といった作業を行った際に，リアルタイムに変更が反映される．

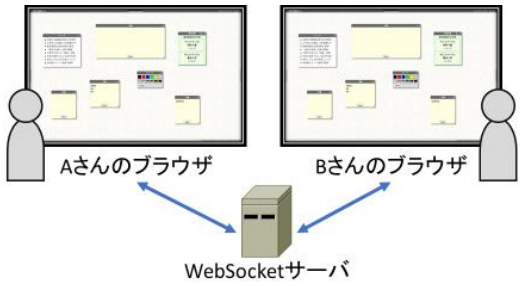

図 1　両ユーザとも同様のデスクトップ画面

また，2 人に限らず更に多い人数で利用する場合も同様に作動する．リアルタイムに他のユーザによる変更点が反映されるため，デスクトップを同時に利用しているということが直観的に理解しやすくなると考えられる．ただし，今回の実装はデスクトップ画面のリアルタイム更新

Copyright ©2015 Information Processing Society of Japan.
All Rights Reserved.

であり，デスクトップ内で作動している各アプリケーション内で変更が行われた場合のリアルタイム更新は未実装である．理由としては，アプリケーションはメインのシステムと切り離されて作動しており，API を利用してメインシステムと連携しているためである．そのため今後，API にリアルタイム通信のための機能を追加し，それを利用してアプリケーションを改良していくことで対応する予定である．

4　システム実装

4.1　Web サーバと WebSocket サーバの共存

本システムでは Web サーバ，WebSocket サーバおよび DB サーバから構成されている(図 2)．Web サーバは静的コンテンツを提供しているだけでなく，PHP を用いたシステムが作動しており，リアルタイムで更新する必要のない変更点に関する情報はこちらに送られ，DB に保存される．一方，リアルタイムに変更点を相手に反映させる必要のある情報は WebSocket サーバに送られる．ただし，この変更点を恒久的なものにするため，および Web サーバとの整合性を保つために，他のユーザに変更点を伝搬させた後，DB に保存する．なお，予めリアルタイムの必要性を考慮してどちらのサーバへ変更点情報を送るかを指定してあるため，同時に 2 台のサーバから DB への書き込みが発生してもデータの競合は発生しない．

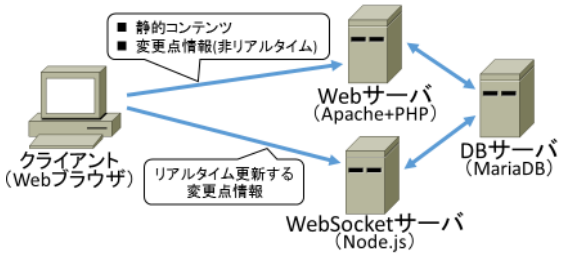

図 2　システム構成

4.2　コネクションの管理

WebSocket を利用するにあたって，Socket.IO と呼ばれるライブラリを使用し，クライアントは JavaScript，サーバは Node.js を利用して構築した．そして，デスクトップの変更に関する情報がブラウザから WebSocket サーバに送られると，他のユーザにその情報が一斉に送信される．しかし，ユーザ A・B・C 間で共有しているデスクトップと，ユーザ D・E 間で共有しているデスクトップが存在する場合，この方法では A・B・C で共有しているデスクトップにおける変更内容が，関係のない D・E にも送られてしまう．そのため，本システムでは WebSocket サーバ内でデスクトップごとにグループを作成し，コネクションをグループ単位で管理することで，関係のない更新情報を送信しないようにした(図 3)．これによりトラフィック量も抑えることができる．

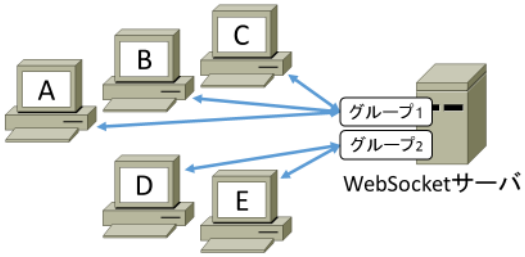

図 3　コネクションはグループ単位で管理

5　おわりに

本論文では WebSocket を実装することでリアルタイム更新を可能にした共有型 Web デスクトップを提案した．リアルタイムを実装していない共有型 Web デスクトップ「AppTop」では，「同時に編集をした際に変更点が反映されず使いづらい」「相手の更新を反映させるためにリロードするのは手間」といったコメントが多くあったが，WebSocket によるリアルタイム更新を実装することで，その問題点は解決できたと考えられる．また，リアルタイム化したことで，デスクトップを複数人で同時に共有しているといった概念を理解しやすくなり，更に活用することができると考えられる．このように共有型 Web デスクトップをリアルタイム化した際の有用性は十分にある．ただし，本システムでは複数人で同時に編集を行った際の排他制御を実装していない．そのため，2 人が同時にウインドウ内で特定のアプリケーションを移動させた場合などに，ユーザ間で操作の取り合いになってしまう．そのため排他制御を実装するほうが望ましいと考えられるが，排他制御を強くしすぎてしまうと，同時に 1 つのデスクトップを編集することができるというメリットが失われてしまうため，どのように制御するのかは慎重に検討する必要がある．今後は排他制御の検討および，アプリケーションがリアルタイム通信を行うための API の整備を行っていく．

参考文献

[1] Netvibes，http://www.netvibes.com/
[2] AstraNOS，http://www.astranos.org/
[3] 鈴木啓真，兼子正勝：グループ共有が可能な Web デスクトップの構築，情報処理学会ワークショップ 2014(GN Workshop 2014)論文集，2014

Copyright ©2015 Information Processing Society of Japan.
All Rights Reserved.