

【様式 4】

テーマ名：ブラウザと統合したHTMLメッセンジャーの開発

申請者名：源馬照明

1. 目的と背景

私の母が、夫婦水入らずの旅行のときにデジタルカメラで撮った写真を20枚、私に送りたいという。

Eメールに20枚の写真を添付すればいいだろう。だが、Yahoo!メールで”ファイルを添付する”ボタンを20回クリックするはめになる。それで私は母にZIP圧縮を説明したが、まったくわかつてもらえなかった。

Yahoo!フォトにアップロードするという手もあるが、Webサービスの使い方を覚えなければいけないし、もしアップロードしたいのが音楽や動画なら、また別のWebサービスを使わなければいけない。
(私でさえ、友人に20曲の音楽ファイルをどう送ったものだかわからない)

結局、母は”ファイルを添付する”ボタンを20回クリックして、私にメールを送ってくれた。その旅行の写真に、息子も爆笑間違いなしの、父の頭のうえにシマリスがのってヒマワリの種を食べている写真があるので、母はメールを*今すぐ*見てほしい。

そこでバッグをゴソゴソ探って、携帯電話を取り出し、息子にかける。

「あ、もしもし。あんた、大学はちゃんと行ってるの？ この前に旅行いったときの写真、メールしたから見なさい。」

「じゃあ、今からメールチェックしてみる。ちょっと待って…。」

「まだ？」

「あ、届いてる。」

「で、その中にシマリスの写真があるでしょ…」

「20個も添付ファイルがあつて、そんなすぐには開けないよ。ちょっと待って。」

「えーと、”この添付ファイルを保存しますか？”はいっと ×20」

「まだ？ 早く」

「あ、コレか！お父さんの上にリスが！あつはつは！」

母は、ただ写真を20枚、私に送って話をしたいだけなのに、

なぜ、20回も”ファイルを添付する”ボタンをクリックしなくちゃいけないんだろう。

なぜ、パソコンがあるのに携帯電話を使わなくちゃいけないんだろう。

テキスト、写真、音楽、動画、これらを簡単にやりとりしたい。つまり、HTMLをやりとりしたい。

必要なのは、携帯電話の気軽さでHTMLをやりとりできる”HTMLメッセンジャー”だ。

2. 提案全体の概要

“人と人をリンクする”
ブラウザ+メッセンジャー

「Musubi」

ブラウザ

メッセンジャー

◆ HTMLの豊かな表現力

テキストとマルチメディアに加え、FlashやJavascriptの動的HTMLで広がる表現力。

◆ ページを結ぶ

ページ間のハイパーリンクが、知識体系の共有を可能にした。

◆ いつも使っている

人は多くの時間をブラウザの前で過ごしている。

× テキストのみ

◆ 人を結ぶ

コミュニケーションしたいという人間の大きな欲求にこたえる。

◆ 対話性

クラウドのシェルとしての可能性。

ブラウザと統合した
HTMLメッセンジャー

M u s u b i

HTMLメッセージ

現状では、MSN MessengerやGoogle Talkなどはテキストメッセージしか送れない。ブラウザとメッセンジャーを統合することで、HTMLメッセージを送れるようになれば、Webを通じたコミュニケーションの表現力は格段に進歩する。Eメールの添付ファイルより気軽にマルチメディアファイルを送れるようにしたい。

人と人をリンクする

1人1人の連絡アドレスをURIにすることで、HTMLで人をハイパーリンクできる。現在、ブラウザでメールアドレスをクリックするとメールが起動するのと同様に、ブラウザに統合したメッセンジャーがシームレスに聞くようになる。ブラウザのブックマークとメッセンジャーのアドレス帳は全く同じ意味になる。

HTMLメッセージ

テキストとマルチメディアによる豊かな表現力

HyperText
= Text
+ Multimedia
+ HyperLink

これで解決！ “母が私に20枚の写真を送るとき”

◆メールの添付ファイルを使う

しかし母はZIP圧縮のやりかたがわからないので、Yahoo!メールの“ファイルを添付する”ボタンを20回クリックするめになる。

◆Yahoo!Fotoにアップロードする

Webサービスの使い方を覚えるのは大変だし、もし音楽や動画だとまた別のWebサービスを使うことになる。

HTMLメッセージなら…

母はパソコンに入っている写真をブラウザで開いて私にメッセージとして送ればよい。フォルダの内容からHTMLを作ってくれるプログラムを使えばまとめて送れて便利。

自動アップロード

```
<html>
<body>



```

パソコンに入ったファイルを、FTPで自動アップロードする。
メールに添付するより簡単である。

HTMLがそのまま届く

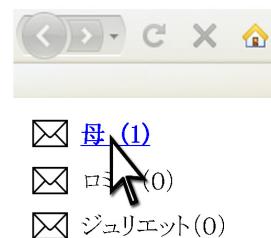
```
<html>
<body>



```

WebのURLに書きかわって届く。
ローカルで確認したHTMLがそのままWebで見られるようになっている。

メッセンジャーに届く



ブラウザで開く

XMPPで届く

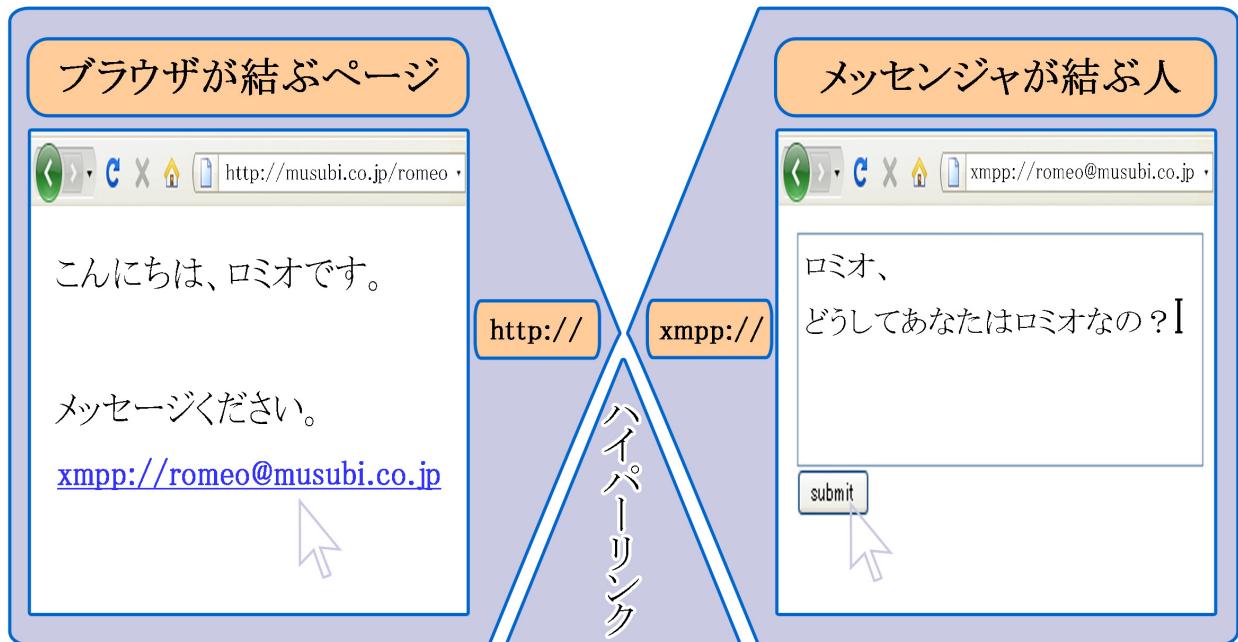
XMPP (eXtensible Messaging and Presence Protocol) は、IETFで標準化している、XMLベースのインスタントメッセージングプロトコルである。Google Talkが採用しているので、ユーザはGMailのアカウントさえあればすぐに参加できる。

Javascriptプログラムも届く

メッセージの作成を補助するようなJavascriptプログラムを送ることもできる。例えば、簡単な入力フォームでアンケートをとるプログラムや将棋の棋譜をメッセージでやりとりしてそれを動的HTMLで盤面に再現するプログラムなど。

人と人がリンクする

HTMLのハイパーテキストで、ページとページを、そして人と人を結ぶ。



1人1人にURIを

xmpp://romeo@musubi.co.jp

ブラウザ拡張

Firefoxブラウザ拡張では、独自のURIスキーマを開発できる。これを利用して“xmpp://”を定義し、XMPP URIへのリンクを開くと、ブラウザに統合したメッセンジャーがシームレスに立ち上がるようになる。

XMPP URI

XMPPのURIアドレスはIETFでRFC5122として標準化している。これからますます広がる可能性が大きい。

エージェント

エージェントとは、人間に代わってメッセージを送受するプログラム。定時刻に天気予報のメッセージを送ったり、メッセージを受け取って集計、分類などを行う。人とコンピュータの新しいつながり。

XUL



XULで実装されたメディアプレーヤー、Songbird。
マルチプラットフォーム(Windows、Linux、Mac OS X)で動作する。
Songbirdはウェブブラウザの機能も持つ。iTunesのライバルを目指し、
オープンソースで開発が続けられている。

XULで実装されたFTPクライアント、FireFTP。

XULで実装されたXMPPクライアント、Sameplace。
XMPPサーバejabberdの開発者Massimiliano Mirraが
開発している。1クリックでインストールでき、GMailの
アカウントさえあれば、すぐに使いはじめることができる。
本提案の競合的存在である。

XULは、ユーザインターフェースを記述するためのXML言語である。FirefoxもXULで実装されていて、マルチプラットフォームなGUIアプリケーションを簡単に開発することができる。またCSS、JavaScript、DOM等の既存の標準技術を多く利用しているため、すでにこれらの技術に親しんでいるプログラマやデザイナにとっては比較的習得しやすい言語である。

XULはリッチインターネットアプリケーション(RIA)の開発基盤として、Adobe AIRやSilverlightとともに最近、注目を集めている。一般的なプログラミング言語はコンパイルの必要があるって編集⇒実行テストのターンアラウンドタイムが大きいのに比べ、Web技術ベースの開発ではほとんどの場合、コンテンツを再読み込みするだけでよいので、頻繁に改良を試すことができ、開発効率が非常に優れている。また世界中の開発者がFirefox開発に力を注いでいるので、開発基盤として安定している。

XMPP

The Extensible Messaging and Presence Protocol PROTOCOL

インスタントメッセージのための
XMLベースプロトコル

XMPPはインスタントメッセージのために開発されたプロトコルで、かつてはJabberプロトコルと呼ばれていたものが、2004年にIETFで、RFC3920として標準化したものである。XMLベースでプロトコルに拡張性があるのが大きな特徴である。HTTPが一方向にプルしかできないのに対して、XMPPは双方向にプル&プッシュでき、対話性がある。

並列分散関数型言語Erlangで
実装されたXMPPサーバ

ejabberdはErlangで実装されたXMPPサーバである。Erlangは、エリクソン社で開発された並列処理指向の関数型言語で、交換機など電気通信システムの開発に使われている。ミッションクリティカル用途のため、リアルタイム性、耐障害性、無停止性に強みがあり、ejabberdの安定性に大いに貢献している。

Firefox拡張のための
XMPPクライアントライブラリ

xmpp4mozは、ejabberdの開発者Massimiliano Mirraによって開発されたFirefox拡張用XMPPクライアントライブラリである。オープンソースで開発されているので、本提案で必要な改善があればこちらにもフィードバックしていくつもりである。

3. 目標と提案内容

テキスト、写真、音楽、動画、これらを簡単にやりとりしたい。つまり、HTMLをやりとりしたい。
必要なのは、携帯電話の気軽さでHTMLをやりとりできる”HTMLメッセンジャ”だ。

そして、そのメッセンジャには、20枚の画像を簡単にアップロードできる機能が必要だ。

それから、私はパソコンを使っているときは常にブラウザを開いていることが多いから、ブラウザにポンッと着信がでて、届いたHTMLメッセージをすぐその場で見られるとよい。

我々に本当に必要なのは、”ブラウザと統合したHTMLメッセンジャ”だ！

3.1 設計、実現方法

- ブラウザと統合するには、Firefoxブラウザ拡張であるXULを用いる。Firefoxブラウザは最近になってシェアを伸ばしてきているし、Firefoxブラウザ拡張はインストールが1クリックで簡単だ。XULはマルチプラットフォームなGUIプログラムの開発基盤で、Firefoxをはじめ、Songbird、Google Gearsなどで使われている。XULを使うと、Adobe AIRやSilverlightのようにRIA(リッチインターネットアプリケーション)を簡単に作ることができる。
- インスタントメッセージングプロトコルには、IETFでRFC3920として標準化しているXMPPプロトコルを用いる。Google TalkもこのXMPPプロトコルを採用しているため、Googleのアカウント(GMailのメールアドレス)さえあればよいというメリットもある。
- XMPPサーバは既にこの分野で実績のあるejabberdを用いるので、私はメッセンジャの開発に集中できる。
- HTMLメッセージの表示はブラウザに任せることができる。
- 自動アップロードについては、FTPプロトコルを用いる。これはごく枯れたプロトコルで、技術者なら簡単にFTPサーバを用意できるはずだ。あるいはAmazon S3を使えば、サーバーマシンを用意する必要すらない。

3.2 新規性

- HTMLメッセージ
HTMLメッセージによって、従来のテキストメッセージと比べて表現力が格段に豊かになる。
- 人と人をハイパーリンクで結ぶ
ブラウザとメッセンジャを統合するにあたって、新しいURIスキーマ、“xmpp://”をブラウザに組み込むことで、人の連絡アドレス(<xmpp://romeo@musubi.co.jp>)にハイパーリンクを張ることができるから、ブラウザでページとページを結び、さらに統合したメッセンジャで人と人を結ぶことができる。
- 人とコンピュータの新しいつながり

人間に代わってメッセージを送受するようなプログラムは、本提案と独立して誰でも開発できる。例えば、定時刻に天気予報のHTMLメッセージを送ってくれるプログラムや、HTMLメッセージを受け取って集計、分類を行ってくれるプログラムを、みんなで作ることができる。

- 動的HTMLメッセージ

HTMLは、テキストとマルチメディアだけではなく、Javascriptのプログラムも含む。そこで、メッセージに、簡単なアンケートを取るためのYes/Noボタンを含んだり、簡単に返信するためのテキスト入力ボックスと送信ボタンを含んだりできる。もっと凝ったプログラムでは、例えば将棋の棋譜をメッセージとして受け取って、それを動的HTMLで盤面に再生してくれるプログラムを使えば、ブラウザ上で簡単に将棋のオンライン対戦ができる。

4. 効果と終了後の展望

4.1 実用化へのシナリオ

開発したFirefoxブラウザ拡張を無料、オープンソースで公開する。

XMPPサーバはGoogle Talkを利用するので用意する必要がない。

FTPサーバははじめこちらで用意するが、技術資料を公開すれば技術者なら用意できると思う。
まずは日本語版を公開し、ボランティアを募って英語をはじめ各国語に翻訳したい。

4.2 展望

メッセンジャーとの統合でブラウザに”対話性”が加わることで、クラウドのシェルとしての、ブラウザの新しい可能性を開くことができる。例えばブラウザで、クラウドで実行したバッチ処理の結果を待つといったことが簡単にできるようになる。

現在のWebサービスには相互に連携する基盤がなく、例えば、GMailからYahoo!Fotoに”コピー＆ペースト”する方法がない。これもHTMLメッセージを基盤とすることで解決できるかもしれない。

UNIXは、人がコンピュータでやりたいことのすべては、テキストファイルの読み書きだという大胆な抽象化で成功した。

本提案では、人がWebでやりたいことのすべては、HTMLファイルの読み書きであり、つまり、HTMLメッセージのやり取りであるという抽象化を試みたい。

- ファイルシステムにHTMLを送れば保存してくれる。
- プリンタにHTMLを送れば印刷してくれる。
- ブログにHTMLを送れば記事を公開してくれる。
- プログラムにHTMLを送れば、集計、分類など、様々な処理をしてくれる。

アラン・ケイがSmalltalkで実現した「メッセージング」による「オブジェクト指向」を、再定義できると思う。最終的には、本提案で開発したHTMLメッセンジャーが、ブラウザに最初から組み込まれるようになるのが理想である。

4.3 競合 Google Talk、MSN Messenger、Sameplace

4.4 競合に対する差別化ポイント

これらは全て、テキストメッセージである。私はHTMLメッセージのほうが優れていると思う。

Google TalkやMSN Messengerには大量のユーザと信頼と実績があるが、最近は新機能の実装などは行われていない。

Sameplaceは、ejabberdの開発者が参加していて、本提案と同じFirefox拡張を使っている点で、注意すべき競合である。ただ、本提案のHTMLメッセージと、"xmpp://"URIスキーマによる人と人のリンクの点では、こちらに優位性がある。

しかし、そもそも、メッセンジャーに一番大切なのはユーザ数、つまり知り合いの誰にでもつながることだ。Google TalkやSameplaceのユーザは、我々のユーザともおしゃべりできるのだから、これらは競合ではなく、共存共栄すべき存在であるとも考えられる。

4.5 ビジネスマodel

ビジネスモデルについてはPMのご指導をいただきたい。

4.6 想定する利用シーン

これまでEメールの添付ファイルを使っていた場面が、そのまま、想定する利用シーンである。

これまで携帯電話のSMSや写メールを使っていた場面もいぐらか置き換えることができるだろう。

4.7 ターゲットする利用者

まずは、普段からFirefoxやGoogle Talkを使っているような技術者や大学生をターゲットとする。そこから徐々に一般向けに裾野を広げたい。

5. 準備状況

機材等についてはすべて準備を終えていて、いつでも開発に入る。

xmpp4mozでブラウザでテキストメッセージをやりとりできることは既に確認している。

また、FTPを使っているブラウザ拡張があることも確認している。

よって、本提案が、技術的に可能であることはわかっている。

- 関連システム: Linux(OS)、ejabberd(XMPPサーバ), ProFTPD(FTPサーバ)
- 開発技術: Firefox拡張(XUL)
- 開発言語: Javascript
- 開発機材: パソコン1台
- ソフトウェアモジュール: xmpp4moz(Firefox拡張用XMPPクライアントライブラリ)

6. 開発計画

"ブラウザと統合したHTMLメッセンジャー"を開発する。具体的には、Firefoxブラウザ拡張(XUL)で、FTPによる自動アップロード機能を持ったXMPPクライアントを、Javascript言語で実装する。

さらにブラウザに"xmpp://"URIスキーマを加えることで、人にハイパーリンクを張れるようにする。

それに加えて、本提案の応用例の実証として、以下を実装する。

- HTMLメッセージの作成を補助するための簡単なJavascriptプログラムとして、ブラウザ上で動くお絵かきチャット。
- 定時刻に天気予報メッセージ送ってくれる程度の簡単なエージェントプログラム。

さらにドキュメントと公式ホームページを制作する。

6.1 進め方

Google Codeにプロジェクトを立ち上げる。Google Code上のプロジェクトWiki、ソースコードのバージョン管理システム、バグトラックシステムを利用する。

6.2 スケジュール

以下を7～10月で行う。

- Firefox拡張用のFTPクライアントライブラリを実装。
- HTMLが含むローカルファイルをFTPで自動アップロードする機能を実装。
- HTMLメッセージの送受実験。

以下を10月～12月で行う。

- "xampp://" URIスキーマを実装。
- HTMLメッセージの一種としてJavascriptプログラムの送受実験。
- Javascriptプログラムの例として、ブラウザ上で動くお絵かきチャットの実装。
- エージェントの例として、定時刻に天気予報メッセージ送ってくれるエージェントの実装。

以下を12月～2月で行う。

- ユーザインターフェースの改良。ドキュメント制作。公式ホームページ制作。
- バージョン1.0公開。公式ホームページ開設。

6.3 開発体制

1人。大学院での研究と並行して開発を進める。

6.4 費用

サービス開始後のFTPサーバレンタル代、インターネット回線帶域代。

7. 提案者の自己紹介

大学院では型理論というプログラミング言語の基礎理論を研究し、趣味でJavascriptでWeb開発をしていて、理論と応用の両方に強みがあります。

プログラミングを始めたのは中学校のときでVisual Basicでした。これまでに、C言語、Scheme、Javascript、OCamlを習得しました。

- Gemmaの日記 <http://d.hatena.ne.jp/Gemma/>
- 2005年12月にソフトウェア開発技術者の資格を取得。
- 2008年に株式会社はてなで、1ヶ月間のインターンに参加し、Webサービスの最前線に触れ、開発体制や運用ノウハウなどを学びました。
- Google App Engineで、ブラウザで動く2chブラウザ”ktkrリーダー”というWebサービスを運用しています。
<http://ktk-reader.appspot.com/>
- 学内のOCaml-Nagoyaという勉強会のメンバーと共に著で、毎日コミュニケーションズ社から「入門OCaml」という書籍を出版しました。
<http://book.mycom.co.jp/book/978-4-8399-2311-2/978-4-8399-2311-2.shtml>
- 技術評論社で「script.aculo.usを読み解く」という、週1回、全13回の連載を執筆しました。
<http://gihyo.jp/dev/feature/01/scriptaculous>
- Shibuya.lispというカンファレンスで自作のSchemeプログラム(FUSEでS式ファイルシステム)を発表しました。
<http://shibuya.lisp-users.org/2009/02/28/sltt-2-tb/>
- Scheme用のXMPPクライアントライブラリを開発しました。
<http://code.google.com/p/gauche-xmpp/>