



## 2008 年度下期未踏 IT 人材発掘・育成事業 採択案件評価書

### 1. 担当PM

筧 捷彦 PM(早稲田大学 基幹理工学部 情報理工学科 教授)

### 2. 採択者氏名

チーフクリエイター: 米倉 健太(早稲田大学理工学部機械工学科)

コクリエイター: なし

### 3. プロジェクト管理組織

株式会社 ゴーガ

### 4. 委託金支払額

2,673,627 円

### 5. テーマ名

Web における front-end —— 『stir』の開発

### 6. 関連Webサイト

なし

### 7. テーマ概要

#### 1. これは何？

web フロントエンド『stir』とは、既存の web の仕組みでは為しえなかった、エンドユーザー主導による積極的な web サービスの利用を実現するシステムです。

「このブログは記事の編集機能は優れているけれど、保存機能がない」「この検索サイトの検索結果がこちらのサイトの製品検索につながればいいのに」といった、web 上に頻繁に存在する既存サービスに対する不満を、ユーザ側から積極的に組み替えて解決してしまおうという概念とアプリケーションフレームワークです。

stir を使用することによって、今までサービスを与えられた範囲内でしか使用できなかったユーザが、サービスをより自分好みに変更したり、新しい機能(「アドオン」と呼ぶ)を付加したりすることが可能になります。

アドオンはその設置されている URI を関連付けることで管理され、各々のユーザがそれらアドオンについて使い心地などのフィードバックを送ることで、他のユーザの評価を参照しながら適切なアドオンを選択できます。

## 2. できること

既存の Web サービスをフレームワーク内に読み込み、外部の任意アドオンを追加すること。

現時点での対応ブラウザは Windows 版 Internet Explorer6,7/Firefox3

## 3. これから追加される機能

- ・ 情報サーバとの連携
- ・ Greasemonkey スクリプトへの対応
- ・ アドオン作成機能

## 4. 将来の展望

自分でカスタマイズした設定をどこからでも参照できるようなログイン制のサービスを提供する。

### (概要)

Ajax を活用して、異なるドメインが提供するサービスを横断的に利用し、さらに独自の機能まで付加しようというのが、この概念とフレームワークの趣旨である。

内部的には javascript の XMLHttpRequest を用いてブラウザの挙動をエミュレートしており、これによって既存 web サイトの 1 レイヤー上の領域を作り出し、そこから web サイトの挙動を操作している。

## 8. 採択理由

インターネット上で提供されている Web サービスに対して、ユーザがその上に様々な機能を追加することができるようにしよう、という提案である。

そのためのフレームワーク stir を開発する。

stir は、Web サービスをフレームワークの中に取り込んで、サーバとクライアント(ブラウザ on ブラウザ)で作業を分担して追加機能を実現する。このとき、サーバ側でや

ることは、プラットフォームによって異なるブラウザの差異を吸収するための作業を分担することにある。

これによって、ユーザが CSS や javascript を使って既存 Web サービスに機能追加して使うことができるようになる。すでにクライアント側のシステムは試作済みというように計画もしっかりしたものであり、Web2.0 と呼ばれているものに新しい地平を開いてくれることを期待する。

## 9. 開発目標

Web ページ上に仮想的なブラウザアプリケーションを作成し、ユーザにはこの仮想ブラウザを通して Web ブラウジングを提供する。この仮想ブラウザは、JavaScript というスクリプト言語を使って記述する。JavaScript は、多くの OS の上のほとんどすべてのブラウザ上で動作する。したがって、こうして作った仮想ブラウザは、インストールの手間などいっさいかけないで、使うことが可能になる。

この仮想ブラウザの設定を Web サービス上に保存することで、ユーザに職場と自宅で同一の設定の Web ブラウジングが提供できる。また、この仮想ブラウザに読み込んだコンテンツをユーザが組み替える機能を実装することで、既存の Web コンテンツに対してユーザが個々人の事情に即して機能を追加できる機能を提供する。

将来的には、ユーザごとのブラウジングの軌跡や設定を集合知的に処理することにより、多数のユーザが選択している設定を他のユーザに提案したり、趣味の似たユーザが好む Web ページを提案したりできるシステムを目指す。

この目的に合わせて、つぎのものを実現する。

### クライアントサイドアプリケーション

- ・ 仮想ブラウザ機能
- ・ ウィンドウ分割機能
- ・ サーバサイドとの連携機能
- ・ コンテンツの組み換え機能
- ・ セキュリティ機能

### サーバサイドアプリケーション

- ・ ユーザ識別機能
- ・ コンテンツとそれに対応する設定の関係管理機能

### デモ・開発用 Web サイト

- ・ デモ・開発用 Web サイト
- ・ 初めて使用するユーザに向けた導入の手引き

## 10. 進捗概要

2月・3月でクライアント(仮想 Web ブラウザ)のユーザインタフェースの設計とプロトタイプを作成するとともに、サーバサイド技術の下調べ(MySQL、php)及びDB設計を行った。4月には Web ブラウザをほぼ仕上げるとともに、公開・配布用サイトを構築し始めた。5月には、コンテンツとアドオンの関係管理を XOOBSで行うことに決め、Web サイトの設計および各ページのレイアウトを設計するとともに、ユーザ用のチュートリアルを書き下した。6月にはユーザに使ってもらって開発したアプリケーションの使用感のアンケートを行い、その結果に基づいてシステムのブラッシュアップを行った。さらに、自己開発したアドオンを登録するシステムの開発、システムのセキュリティ対策などを行い、残り時間をシステムの改善作業に当てた。

## 11. 成果

開発したシステムは、改めて Systir と名付けられた。計画当初、このシステムが各種の機能を“攪拌”して表示する、という意味をもたせて stir と名付ける予定であった。ただ、この単語の英語発音は日本語になじまないため、“シスター”と発音させるべく Systir(System + stir)と綴ることにしたという。

さて、その Systir 仮想ブラウザのユーザインタフェースは、図 1 のようである。用意された Web サイトにアクセスするだけで、自動的にこの画面になる。



図 1 仮想ブラウザの画面表示例

画面はいくつかの窓に分割されている。図 1 は、最初に Systir サイトにアクセスしたときの画面配置であり、左の窓で Google の検索ができ（図では“test”で検索した結果が表示されている）、右上の窓には Systir サイトのチュートリアルページが表示されている。右下の窓は、すでに Google で別の検索を行った結果が表示されている。この窓の分割は、全体窓の右上に並ぶアイコンをクリックすることでその分割の方式を変更することができる。横に分割・縦に分割という定番から、タブ形式・右側だけタブ形式・下側だけタブ形式といった変則的な分割方式までが用意されている。

それぞれの窓は、ユーザと Web コンテンツとの間で薄いフィルタのように動作し、ユーザが特殊な操作をしないで各種の追加機能・拡張機能が働いてくれる。その結果、「左側のウィンドウで開いた英文のニュースサイトでマークした英語の単語を、右側のウィンドウで開いた翻訳サイトで瞬時に和訳する」といった、分割ウィンドウ間でデータのやりとりをして挙動を変化させるような使い方もできるようになっている。

こうした追加機能・拡張機能を付加導入する仕掛けは、Firefox の Greasemonkey など様々流布している。こうした追加機能・拡張機能を使うには、使っているブラウザとその環境それぞれにインストールする（アドオンする）必要がある。しかも、それらがアップデートされることがあるので、常にそうしたアップデートに注意して対応して再インストールする必要がある。

Systir では、その働きを Web サービスとして吸収しているの、個々のユーザは

Systir に対してどのサービスにどの追加機能・拡張機能を適用したいかを登録するだけで、使う PC が変わってもブラウザが変わっても何も心配する必要がない。

Systir に用意されている追加機能・拡張機能は、それらを管理する特定ページ ([http://www.systir.net/addon\\_regist/addon\\_manage.php](http://www.systir.net/addon_regist/addon_manage.php)) にアクセスすることで行うことができる(図 2)。

Open new tab	Ver.01.0	リンクを新しいタブで開きます	<input type="checkbox"/>
さんぶるあどおん	Ver.00.1	これはテスト用のサンプルアドオンです。	<input type="checkbox"/>
test addon	Ver.00.1		<input type="checkbox"/>
mixi_addon	Ver.01.0	mixi用アドオン	<input type="checkbox"/>
Google Search Sidebar	Ver.1.6.4	Google用アドオン	<input type="checkbox"/>
Google Image Search Advanced	Ver.1.4		<input type="checkbox"/>
CiNii検索結果表示	Ver.1.0		<input type="checkbox"/>
GoogleSearchAddThumbrail	Ver.00.4		<input type="checkbox"/>

ターゲットURL =

アドオン名	バージョン	ターゲットURL
systir用アドオン	Ver.01.0	systir.net*
Open new tab	Ver.01.0	www.google.co.jp/search
Open new tab	Ver.01.0	search.live.com/results.aspx
systir用アドオン	Ver.01.0	www.systir.net*

図 2 適用する追加機能を Web コンテンツごとに設定する画面

このように、追加機能・拡張機能の自由度を上げると、セキュリティの面で脆弱になる。Systir も同じ問題を抱えることになった。そこで、Systir では、追加機能からアクセスできる情報を追加機能毎にユーザが“セキュアレベル”を設定することにして解決している。この“セキュアレベル”というのは、Systir 内で追加機能が作動するとき動作し、そのアクセスできる範囲を“0-すべて許可、1-同一ドメインまで許可、2-同一ドメインの同階層まで許可、3-すべて不許可”に制御する。この“セキュアレベル”は追加機能の作者が Systir にその追加機能を登録する際に推奨値を指定し、ユーザはそれを参考にして自分のシステムでの許可範囲を設定する方式としている。

こうしたユーザ個別の設定が保持できるように、Systir はユーザ管理機能も備え、データベースも置いて管理している。図 3 に示すとおり、Systir での“コンテンツ⇔追加機能・拡張機能”の対応関係は、ネットワーク状につけることができる。

## systir とコンテンツと追加機能の関係

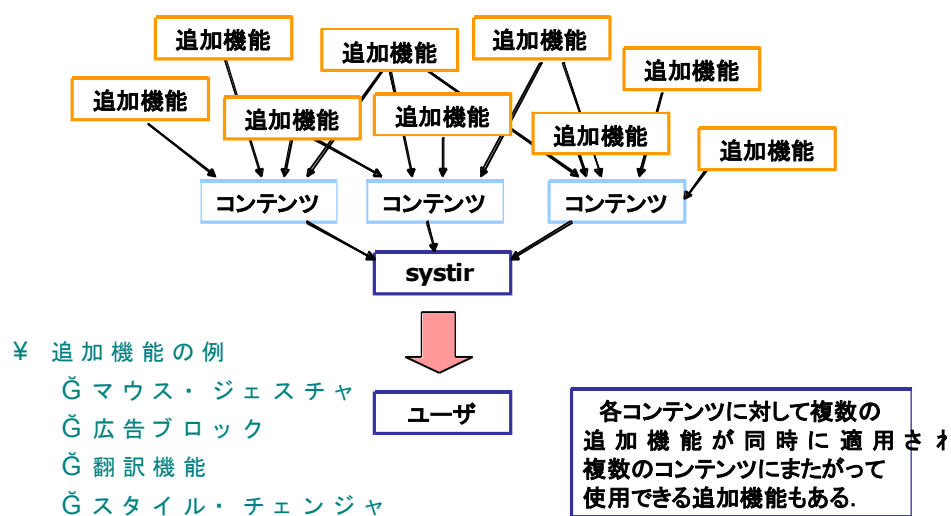


図 3 Systir 下の“コンテンツ⇔追加機能”対応

クリエイターは、Systir のデモを体験できる Web サイト (<http://www.systir.net/>) を用意している (図 4)。そこには、デモを体験した人からの意見を受け取れるように、アンケート機能も用意している。このページは、また、Systir のクリエイター、追加機能・拡張機能の開発者そしてユーザが互いに交流を図るための掲示板も備えている。



図 4 Systir の Web サイト (<http://www.systir.net/>)

Greasemonkey API への Systir への組込み状況は、つぎに示すとおりである。

組込み	Greasemonkey
状況	API
○	GM_log
○	GM_setvalue
○	GM_getValue
○	GM_xmlHttpRequest
×	GM_addStyle
×	GM-openInTab
×	GM_regisMenuCommand



また、Systirと同様の機能提供を行うGreasemonkey、iGoogleと特性を比較すると図5のようになる。

	Greasemonkey	iGoogle	stir
環境依存性	×	◎	◎
初心者の導入の容易さ	△	△	◎
設定の共有	×	◎	◎
追加機能の組み合わせ	◎	×	◎

◎ ◎      △ △  
○ ○      × ×

図5 Systirと類似システムの特性比較

## 12. プロジェクト評価

クリエイターは、理工学部大学院に籍をおくものの、情報が専門の学科・専攻ではない。それでも、未踏ユースの短い開発期間の中でこれだけのものを作り上げてくるのだから、いかに開発環境が整い、ライブラリなどがネット越しに容易に利用できるとはいえ、そのソフトウェア開発力はたいしたものである。プロジェクトの進行も、無理のない形で開発が進んでいるのだから特に不足な点は見当たらない。

ただ、提案し取り組んだテーマ自体が、どれほど多くのユーザが集められるか、使ってもらえるかが勝負を分けるものだっただけに、普及活動、特に熱心なユーザ獲得に知恵をしばり、実践する余裕がもてなかったことが残念である。内容的にどんなすばらしい環境であったとしても、それを使ってくれる人がいてくれないことには絵に描いた餅に終わってしまう。

仮想ブラウザはできた。ユーザ管理をするシステムも用意した。そこでのセキュリティ対策もそれなりに施した。使ってもらうためのチュートリアルも用意したし、サーバも用意した。ついでに使ってみてのアンケートをもらうための仕掛けも用意した。道具立てとしては必要なものを皆そろえた。それで何が足りないのか？

時間が足りなかった。それもある。こうしたシステムがユーザに受け入れられ、使う人が増えていくには、それなりに時間が必要であることは否めない。

宣伝も足りなかった。ウェブサイトを用意しただけで人が集まってきてくれるとは限らない。それなりに宣伝活動を行う必要がある。

しかし、なんととっても、このシステムを使いたい、使うとこんないいことがあるよ、という実際的な例が示せなかったことが大きい。つまり、このシステムに対するキラーアプリケーションが欲しいのである。

それを考え、必要なら編み出して、せつかく作ったこのシステムを広めてほしいものである。

### 13. 今後の課題

クリエイター自身も認めている通り、このシステムを使ってくれる人たちの輪を生み出し、広げていくことがなんととっても大きな課題である。そのためにも、まず、Systir を使うことで便利に使える(使わなければ便利には使えない)例を - そうした Web サービスを、作ること(見つけ出して示すこと)が何より大事である。宣伝活動も、そうした例を挙げて行えるかどうかで、ユーザの獲得が大きく進むかどうかが変わってしまう。ミドルウェアを作ることも大変だけれど、それを普及することはもっと大変である。そして、ミドルウェアといえども、いや、ミドルウェアだからこそ、使ってもらえないことには話が始まらない。是非、その持てる力をそうした例作りに注いでほしい。