

九五八らぼ - KeySnail Scripts

<http://sites.google.com/site/958site/Home/keysnailscript>

KeySnail プラグインを精力的に開発されている 958 氏のプラグイン紹介ページです。モダンな機能の駆使されたソースコードは、プラグインを開発する際の参考になることかと思えます。

まとめ

本稿では設定ファイルやコマンドシステムをはじめとする、KeySnail の応用的な機能について説明しました。KeySnail が皆さんの Firefox を手軽に、そしてディープにカスタマイズしてくための十徳ナイフとなることを祈っています。

— Masafumi Oyamada



HACK
#14

Vimperator と KeySnail の設計上の違い

2 つとも似たような拡張機能ですが、方法論は異なります。

Vimperator と KeySnail

[Hack #10] から [Hack #13] では Vimperator と KeySnail の紹介をしました。両者とも Unix OS の代表的なエディタを模して作られてきた拡張機能で、キーバインドやコマンド、コマンドラインなど多くの共通点を持ちます。

しかし、その実現方法や思想にはやや異なるものがあります。同じ機能でも、作者の思いによってその実装方法は異なってくるものです。その違いを Vimperator と KeySnail を通して見てみましょう。

基本思想

Vimperator は vim、KeySnail は Emacs を模して作られており、もちろんその違いがあるのですが、それ以外にも思想の違いがあります。

Vimperator はほぼ vim 的な部分を前面に出し、ユーザーにもある程度それを強いています。一方で KeySnail はユーザーに合わせてキーバインドの初期設定を選べるようにし、Emacs であることを特に強制していません。これは、多様な人に合わせて選択肢が用意しようという考えが表れたものといえます。

もちろん Vimperator でもキーバインド等をカスタマイズすれば如何様にもできますが、初期導入段階において選択肢が用意されているのといないのとでは、大きな違いがあるのではないのでしょうか。

HTML対XUL

UIの構成技術にも Vimperatorと KeySnailの設計思想の違いが表れています。まずは、2つの拡張機能におけるコマンド補完のUIに注目してみましょう(図14-1、図14-2)。

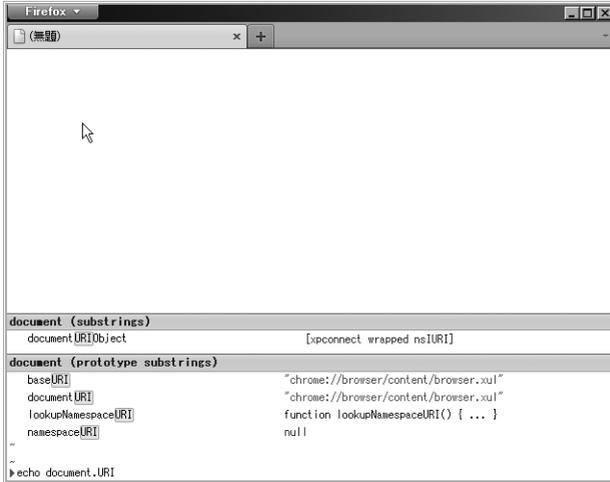


図14-1 Vimperatorの補完UI

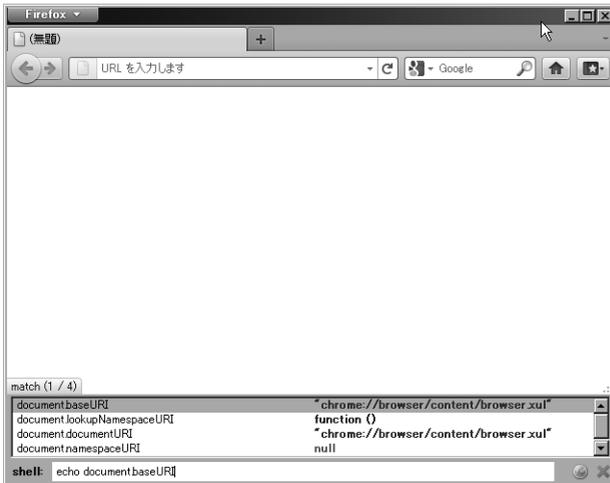


図14-2 KeySnailの補完UI

Vimperatorの補完UIがiframeにHTMLを読み込ませて実装されているのに対して、KeySnailの補完はXULのlistboxで実装されています。

XULはFirefoxのUIに使用されているXML言語で、OSなどのプラットフォームに合わせたUIを形成することのできるマークアップ言語です。UI作成用の言語ですので、リストボックスやツリー、ボタンなどさまざまなコントロールボックスを持ち、表示もOSの標準に合わせてられるため、矛盾のないUIを形成することができます。その代わりに、HTMLの持つようなインライン要素に乏しく、細かなデザインは苦手という側面を持っています。

一方、HTMLは文書作成用のマークアップ言語であり、多くのインライン要素を持つため細かな制御が可能です。しかしUIの作成には不向きな面があり、多くのコントロールボックスを独自にコーディングする必要があるため、コードが煩雑になる傾向があります。

また、XULと違いHTMLは一部を除きスタイルや挙動がプラットフォームを問わず同じであるという特徴があります。これは、場合によってはOS標準のUIと矛盾するため、ユーザーを混乱させる可能性があります。例えば、ダイアログボックスの[OK]や[キャンセル]のボタンの位置を見てみると、Windowsでは[OK]が左、[キャンセル]が右に位置され、GNU/Linuxでは逆になります。これをHTMLで書くとOSを検出し条件分けして書く必要があるでしょう。一方、XULでは労せず標準のダイアログを出せばよいだけとなります(図14-3、図14-4)。



図14-3 Windowsの確認ダイアログ



図14-4 GNU/Linuxの確認ダイアログ

Vimperatorではvimを模倣してhighlightコマンドにより表示されるアイテムなどの見た目をCSSで変えられるようにする機能があります。この機能を実現するためには、細かく制御できるHTMLのほうがよいと判断されたのだと思われます。

設定ファイル

設定ファイルからも、両者の設計思想が透けて見えてきます。

VimperatorとKeySnailはどちらも独自の設定ファイルを持ち、それを読み込むことで自分に合ったカスタマイズが行えるようになっていきます。両者共にUnix文化を反映しており、ホームディレクトリなどに配置された設定ファイルを好みのエディタで編集することができます。

しかしながら、その記法は両者で大きく異なります。

Vimperatorでは設定ファイルはvimに似せた記法で書くことになっています。Firefoxの起動時に1行ずつ読み取られ、それがコマンドとして実行されるイメージです。「コマンド」ですので、Vimperatorのコマンドラインに記述する内容と同じ書き方となります。そのため、コマンドラインから試し打ちをし、気に入った設定ができたらそれをそのまま設定ファイルに記述することが可能です。

一方、KeySnailの設定ファイルは独自の書式を持たず、JavaScriptで記述するようになっていきます。「Emacsを意識して設計した拡張機能であるのに、どうしてLispで書けるようにしなかったのか?」と問われることがしばしばあるのですが、これには2つの理由があります。

1つ目の理由は「JavaScriptを用いた場合、実装が単純かつ高速となる」というものです。ブラウザであるFirefoxにはJavaScriptの処理系が組み込まれており、拡張機能から任意のスクリプトを読み込んで実行することができます[†]。そのため設定ファイルがJavaScriptとして記述されていれば、この機能を使ってその設定ファイルを読み込むだけで設定が行えます。これに対し、LispやVimスクリプトのような記法を導入する場合にはJavaScriptによって構文解析などの処理を実装する必要がおそらくは出てきますが、実装に労力が必要ですし、ネイティブ実装と比較すれば速度的に劣ったものになってしまうことでしょう。

2つ目の理由は「独自の記法は対象ユーザーを限定しうる」というものです。確かにLispによる設定は、一部のユーザーにとって非常に魅力的なことでしょう。しかし、多くのユーザーが抵抗を覚えるであろうことも事実です。FirefoxをはじめとするWebブラウザでは長らくJavaScriptが用いられてきました。そのためJavaScriptを読み書きすることができるユーザーは少なくありません。そうした設定ファイルの編集自体に抵抗を覚えるユーザーのためにGUIによる設定方式も用意しました。これにより、幅広いユーザー層をカバーできるようになりました。

[†] eval関数やFunctionコンストラクタに加え、mozJSSubScriptLoaderやComponents.utils.importなどが利用できます。

まとめ

本稿では、いくつかの側面から Vimperator と KeySnail の設計上の違いについて見てきました。どちらもよく似た機能を提供する一方で、その設計思想に大きな違いがあることがわかっていただけたと思います。「どうしてこんなつくりになっているのだろう」という疑問を抱いたときには、制作者の込めた思いを想像してみると面白いかもしれません。

— Makoto Terada & Masafumi Oyamada



Firefox と Twitter

Twitter で使える拡張機能やユーザースクリプトを紹介します。

Twitter (<http://twitter.com/>) は今や日本において確固たる地位[†]を築いた Web サービスです。Firefox では多くのアドオンや、Greasemonkey のユーザースクリプトによって快適に使えるようになっています。

Twitter Address Bar Search

Twitter Address Bar Search (<https://twitter.com/#!/download/firefox/>) は Twitter 社による公式の拡張機能です。この拡張機能は Mozilla Labs で開発された AwesomeBar HD (<http://mozillalabs.com/prospector/2011/04/27/awesomebar-hd-find-what-you-want/>) のアイデアを元に開発されています。

AwesomeBar HD はアドレスバーと検索バーを統合し、アドレスバーの機能を強化したり、検索のカテゴリを入力することで検索方法を分けたり、検索のカテゴリを素早く切り替える仕組みを提供しています。例えば、統合されたバーで「search: firefox」と入力するとデフォルトの検索エンジンを用いて検索をし、「book: firefox」と入力すると Amazon による検索を行います (図 15-1)。



図 15-1 AwesomeBar HD の動作

[†] 有名脚本家によるドラマなんかもありましたね。