

---

## インターフェイスの街角 (61)– WikiHelp

増井 俊之

---

---

### Wiki の活用法

2001 年 4 月号や 12 月号で、Web 上での情報交換に便利な “Wiki Wiki Web (Wiki)” を紹介しました。その後、“The Wiki Way” [1] のような書籍が出版されるなど、Wiki は徐々に広まりつつあります。しかし、まだまだ多くの人に活用されているとは言いがたいようです。

Wiki が思ったほど活用されていないのは、なんとなくそれ自体の世界で閉じているように感じられるからかもしれません。Wiki ページどうしをリンクしたり、Wiki ページから外部の Web ページを参照するのは簡単ですが、Wiki とは関係のない世界から Wiki ページを参照しようとすると、意外なほど手間がかかります。

たとえば、<http://wiki.example.com/> というサイトの abcde という Wiki ページを参照するには、

`http://wiki.example.com/wiki?abcde`

という URL を指定する必要があります。こういった長い URL をいちいち入力するのは大変なので、ついつい、ブックマークなどに登録した Wiki のトップページからリンクをたどって abcde のページを参照する方法が基本になってしまいます。つまり、Web ブラウザ以外の手段では、それほど手軽に参照できないわけです。

一方、Wiki ページはさまざまな情報の交換に適しているので、シェルのコマンドラインなどからも参照できれば重宝することがあります。たとえば、UNIX のあるコマンドにどんな引数が指定できるのかを忘れてしまった場合、そのコマンドの使用法を書いた Web ページをすぐに見られれば便利です。また、コマンドの使い方についてほかの人と情報の交換ができれば、さらに役立つでしょう。

現在、複雑なシステムの利用法を解説する Web ページの多くは開発者によって用意されています。一方、ユーザー間の情報交換には、メーリングリストや掲示板が使われることが多いようです。しかし Wiki を使えば、マニュアル的なページも情報交換のページも同じ場所に置けますし、マニュアルの誤りなどをユーザーが訂正することも可能です。また、システムが日くなってくると、開発者は Web ページの整備にあまり熱意を示さなくなる傾向があります。そのような場合も、Wiki ページならユーザーが自由に管理して情報を充実させることができます。

Wiki ページは、さまざまな場面で有用な情報の提供や交換に利用できるはずですが、マニュアルやヘルプがあまり整備されていないシステムは珍しくありませんが、Wiki を使う人がさらに増えれば、ユーザーの努力や情報交換によって徐々に内容が充実していく可能性もあります。

今回は、各種のシステムのマニュアルやヘルプを Wiki ページで簡単に扱う方法を考えてみます。

---

### Wiki ページを作る

まず、ヘルプや情報交換のための Wiki サイトを用意します<sup>1</sup>。今回は、2001 年 12 月号で紹介したような、Wiki と掲示板を融合したシステムを使うことにします。用意した Wiki ページのリストは、図 1 のようになります。

#### man ページの作成

man ページはあっても Web ページはないコマンドについては、man 形式のファイルを HTML 形式に変換し、それを Wiki ページとして利用すればよいでしょう。

---

<sup>1</sup> <http://www.csl.sony.co.jp/person/masui/WikiHelp/>

図 1 ヘルプのための Wiki ページ



既存の man ページを HTML 形式にするには、`man-2html` と呼ばれるツールを使えばいいでしょう。

man2html には、いくつかのバリエーションがあります。たとえば、手許にある FreeBSD マシンには Earl Hood 氏が作成したもの<sup>2</sup>がインストールされていました。Red Hat などの Linux システムには man パッケージの一部として Andries Brouwer 氏が作成した VH-Man2html<sup>3</sup> というバージョンが入っているようです。前者は、`roff -man` の出力を変換して HTML ファイルを生成します。一方、後者は man 形式のテキストを直接 HTML に変換するため、よりきめの細かい HTML ファイルが得られます。

Linux の JM Project<sup>4</sup> では、この man2html を用いて UNIX のマニュアルの日本語訳を HTML 化する作業をおこなっています。通常の man2html は日本語に対応していませんが、man-1.5g.tar.gz<sup>5</sup> に対して日本語化パッチ<sup>6</sup> を当てると日本語が使えるようになります。

これを使って grep のマニュアルを Wiki ページにした例を図 2 に示します。

## コマンドラインからの呼出し

Wiki Wiki Web の本家である Cunningham & Cun-

2 <http://www.oac.uci.edu/indiv/ehood/man2html.html>

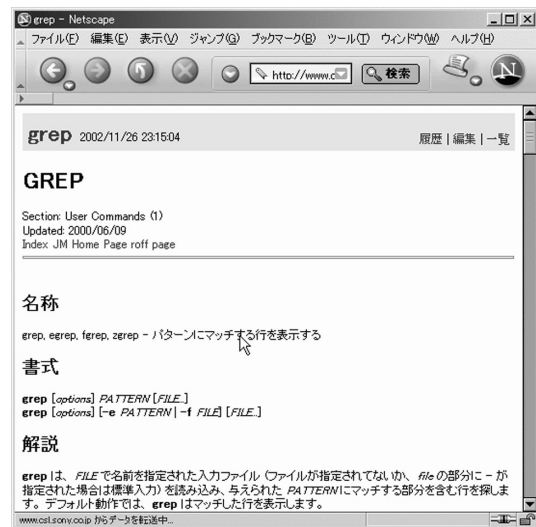
3 <http://users.actrix.co.nz/michael/vhman2html.html>

4 <http://www.linux.or.jp/JM/>

5 <ftp://ftp.win.tue.nl/pub/linux-local/utlils/man/man-1.5g.tar.gz>

6 <http://www.linux.or.jp/JM/man-ja/man-1.5g-ja2.diff.gz>

図 2 grep のマニュアルページ



ningham の `WelcomeVisitors` のページは、

<http://c2.com/cgi/wiki?WelcomeVisitors>

という URL でアクセスするようになっています。その他の Wiki システムでも、たいていはこれとほぼ同じ方式で各ページにアクセスします。つまり、URL の一部に Wiki ページの名前を指定すれば、その Wiki ページが直接呼び出される仕組みになっているわけです。

したがって、コマンドラインで指定した引数にもとづいて、

<http://c2.com/cgi/wiki?Wikiページ名>

といった形式の URL を生成し、これを Web ブラウザに渡して起動すればいいことになります。

### コマンドラインからのブラウザ起動

最近のブラウザは、コマンドラインから簡単に Web ページを呼び出せるようになっています。

たとえば、Linux や FreeBSD などでは Netscape や Mozilla を利用しているのなら、一般的なコマンドの場合と同様にシェルから、

```
% netscape www.google.com
```

のように URL を引数に指定して Web ブラウザを呼び出します。すると、Web ブラウザが起動してそのページが表示されます。ブラウザがすでに起動されているときは、

図 3 Cygwin 用の wiki コマンド

```
#!/bin/sh
BROWSER="/cygdrive/c/Program Files/Netscape/Netscape/Netscp.exe" # Netscape
# BROWSER="/cygdrive/c/Program Files/Internet Explorer/IEXPLORE.EXE" # Internet Explorer
CGI="http://www.csl.sony.co.jp/person/masui/WikiHelp/programs/search.cgi"
"$BROWSER" "$CGI?title=$1"
```

指定した URL のページが別のウィンドウとして表示されます<sup>7</sup>。

Mac OS X では、open コマンド<sup>8</sup>を使い、下記のように引数として URL を指定すれば、指定した URL のページをデフォルトの Web ブラウザで表示させることができます。

```
% open www.google.com
```

Mac OS X のターミナル上で Wiki ページの名前を指定して表示させるには、以下のような簡単なプログラムを使えばいいでしょう。

```
% cat c2
#!/bin/sh
open "http://c2.com/cgi/wiki?${1}"
% ./c2 WelcomeVisitors
ブラウザでWikiページを表示
%
% cat wiki
#!/bin/sh
open "http://www.csl.sony.co.jp/person/masui/WikiHelp/programs/search.cgi?title=${1}"
% ./wiki grep
WikiHelpのgrepのページを表示
%
```

Windows で Cygwin を使っているのなら、図 3 のようにすればよいでしょう。

### 既存のプログラムからの呼出し

図 3 のようなプログラムを使うと、

```
% wiki grep
```

7 Netscape 4.x では、`netscape -remote 'openURL(URL)'" のように指定すれば、すでに起動されている Web ブラウザ上に URL のページが表示されます。

8 open は NEXTSTEP の時代からあったコマンドで、アプリケーションやファイルのアイコンをユーザーがダブルクリックしたのと同様同じ処理がおこなわれます。

といった指定で `grep` という名前の Wiki ページを表示できます。マニュアルページを前提とするならば、

```
% grep --wiki
```

のように、コマンド名のオプションとして指定して Wiki ページを呼び出す方式のほうが直感的かもしれません。

通常、コマンドライン・オプションを追加するにはコマンドを作りなおす必要があります。しかし、最近の UNIX はライブラリ置換機能をもつものが多いので、コマンド引数を解析するライブラリを `--wiki` を解釈するものに置き換えてしまえば、そのライブラリを使うあらゆるコマンドで `--wiki` オプションが使えるようになります。

具体的には、ほとんどのコマンドは getopt(3) を利用しているので、`--wiki` を解釈する getopt() を用意し、`--wiki` が指定されたら前述の wiki プログラムを呼び出すようにすればよいでしょう

### Mac OS X のライブラリの置換え

Mac OS X では、DYLD\_INSERT\_LIBRARIES という環境変数を指定すれば、使用するライブラリを置き換えることができます。

まず、getopt() 関数に手を加え、`--wiki` オプションが指定されたら wiki コマンドを呼び出すようにしておきます。そして、図 4 の Makefile で共有ライブラリを作成すれば、以下のようにして `--wiki` オプションを有効にすることができます(誌面の都合上、⇒ で折り返しています)

```
% setenv DYLD_INSERT_LIBRARIES ⇒
libwikigetopt.dylib
% setenv DYLD_FORCE_FLAT_NAMESPACE YES
% grep --wiki
grepコマンドのWikiページを表示
%
```

### Emacs からの呼出し

コマンドラインからだけでなく、GUI アプリケーシ

図 4 Mac OS X の共有ライブラリを作成する Makefile

```
#
# "--wiki" を解釈する getopt 共有ライブラリの生成
#
libs: getopt.o getopt1.o
gcc -dynamiclib -install_name /usr/local/lib/libwikigetopt.1.dylib \
-compatibility_version 1.0 -current_version 1.0.1 \
-o libwikigetopt.1.0.1.dylib getopt.o getopt1.o
rm -f libwikigetopt.1.dylib libwikigetopt.dylib
ln -s libwikigetopt.1.0.1.dylib libwikigetopt.1.dylib
ln -s libwikigetopt.1.0.1.dylib libwikigetopt.dylib

getopt.o: getopt.c
gcc -c -fno-common getopt.c
getopt1.o: getopt1.c
gcc -c -fno-common getopt1.c
```

図 5 wikihelp.el

```
;;
;; wikihelp.el: EmacsのメニューからWikiを呼び出す
;;

(defvar wiki-browser "/Program Files/Netscape/Netscape/Netscp.exe")
(defvar wiki-cgi "http://www.csl.sony.co.jp/person/masui/WikiHelp/programs/search.cgi")

(defun wiki-call-browser (title)
  (start-process "WikiHelp" nil wiki-browser
    (concat wiki-cgi "?title=" title))
  )

(defun wiki-html-header ()
  (interactive)
  (wiki-call-browser "HTML header"))

(defun wiki-html-css ()
  (interactive)
  (wiki-call-browser "CSS"))

(easy-menu-define
 dummy-symbol html-mode-map "dummy-doc"
 '("Wiki"
   ["ヘッダ" wiki-html-header t]
   ["CSS" wiki-html-css t]
  ))
```

ョンでヘルプ機能を選択したときも Wiki ページが呼び出せればさらに便利です。

GUI プログラムのメニューが拡張可能かどうかはアプリケーションごとに異なります。たとえば、Emacs ではユーザーがメニューを定義できるので、Wiki ページを参照するようなヘルプメニューも作れます。

HTML モードでヘルプファイルを表示し、Wiki ページにジャンプさせたいのなら、easy-menu-define 関数を使って図 5 のようにメニューを定義すればいいでしょう。

このように設定しておくと、Emacs で HTML モードに入ったときに “Wiki” というメニューが表示され(図 6) “CSS” などのメニュー項目が選択できるようになります。

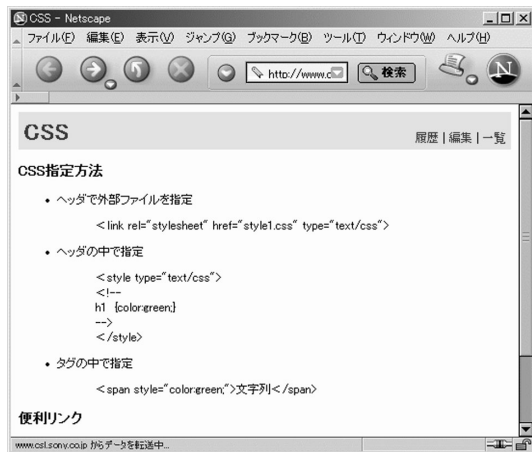
図 6 の画面で “CSS” を選択すると、CSS のページが表示されます(図 7)。

Emacs には、それ自体の機能に関する情報は “ヘルプ” や “info” のように内部で完結しているという特徴があります。しかし、この例のように Emacs とは直接関係のな

図 6 Emacs からのメニュー呼出し



図 7 Emacs から呼び出した CSS のページ



い情報にアクセスする場合は、Web ブラウザを使って参照するほうが便利かもしれません。

## WikiHelp システムの効用

発表されたばかりのプログラムの場合は、たいてい専用の Web ページが用意され、サポート用のメーリングリストや掲示板が運用されていることも多いようです。

しかし、いくらプログラムが優れていても、その作者が読みやすい Web ページを作れるとはかぎりません。とくに、初めて見たプログラムについて調べる場合などは、Web ページの解説を読んでもよく分からないことがあります。私自身も、しばしば以下のような問題に悩まされます。

- そもそも、何をするためのプログラムなのかが分からない。
- ほかの同種のシステムとの違いがよく分からない。
- バグをみつけたときなどの連絡先が明記されていない。
- 関連する情報がどこにあるかが分からない。

図 8 qsort の man ページ



分かりにくい Web ページにいきあたり、ページに誤りを発見したときなどは、ついつい修正したくなります。また、要望などを書き込みたいこともあります。通常の Web ページには書き込みを受け付ける機能はありませんが、Web ページが Wiki になっていればこれらの問題は解決します。

たとえば、qsort(3) は使用法がひどく分かりにくい関数の 1 つで、図 8 のようなマニュアルページを読んでも、どのように使えばいいのかがいまいひとつ分かりません。

しかし、この素っ気ない説明に疑問を感じたユーザーが図 9 のような“使用例”を追加する仕組みがあれば、もうすこし簡単に使えるようになるかもしれません<sup>9</sup>。

新たなプログラムを開発して公開するとき、最初からマニュアルや掲示板などをきちんと揃えた Web ページを用意するのはかなり面倒なものです。しかし、Wiki を利用すれば、とりあえず最小限の情報だけを載せておき、反響に応じて順次整備していくような方式も考えられます。

私自身も、Web が普及する前に作ったプログラムについては、なんとなく億劫な気がしてほとんど Web ページは作っていませんでした。しかし、今回紹介したような Wiki ページを利用する方法であれば、もっと気軽に対応できそうです。

<sup>9</sup> 間違った使用例が追加されると、それはそれで困ったことになってしまいますが、ユーザー間での相互チェックがうまく働けばこのような問題は解決すると思います。

図 9 使用例を追加した qsort の man ページ



```
でいいよ。

返り値
qsrt0は値を返さない。

使用例
なんで man にサンプルコードないんじや! 激怒
qsrt 関数の使い方はむしろ詳しいんだよ。がモロとこ。

int
scmp (const void *ptr1, const void *ptr2)
{
    const char *str1 = *(const char **)ptr1;
    const char *str2 = *(const char **)ptr2;

    return strcmp(str1, str2);
}

int
main (int argc, char **argv)
{
    int i;
    qsort(argv, argc, sizeof(char *), scmp);
    for (i = 1; i < argc; i++)
        printf("%s\n", argv[i]);
}

/*
 * 実行結果
 * $ ./a.out foo bar baz quax
 * bar
 * baz
 * foo
 * quax
 */
www.csl.com.jp 30% データ転送中
```

## おわりに

大がかりなプログラムのマニュアルを読む場合、文字端末上で man コマンドを使い、上を見たり下を見たりする方式は、どうも時代にそぐわない感じがします。たとえば、`man perl` と入力すると、関連するコマンドのマニュアルページ一覧が表示されるだけなどというのは、不親切の極みといえるのではないのでしょうか。端末上で利用する man ページには、別の情報へのリンクが設定されているわけではなく、使用例もあまり載っていないのが普通です。もちろん、ユーザーどうして情報を交換することもできません。Wiki を使うかどうかはともかく、今後はヘルプやマニュアルは Web 上の情報を参照するほうが一般的になるかもしれません。

最近は、検索などのサービスを提供する Web ページをコマンドラインから呼び出すツールも増えてきているようです<sup>10</sup>。Web ブラウザの GUI とコマンドラインの長所をうまく組み合わせた方法を工夫すれば、さらに便利になるでしょう。

(ますい・としゆき ソニー CSL)

### [参考文献]

- [1] Bo Leuf and Ward Cunningham, *The Wiki Way*, Addison-Wesley, 2001 (邦訳: 『Wiki Way』、ソフトバンク、2002年)

<sup>10</sup> <http://surfraw.sourceforge.net/>など