

Web Server の CGI を使った機能拡張

高 嶋 隆 一

(京都教育大学)

Extra Functionality of Web Server by Using CGI

Ryuichi TAKASHIMA

1998年9月30日受理

抄録：京都教育大学の理学科において Web サーバー natsci が運用されている。[1]

Web サーバーは外部に情報を提供するだけでなく、外部から情報を受けとる機能が重要である。情報のやりとりを行なう Web の機能は Common Gateway Interface と呼ばれるものである。これにより Web サーバーをデータベースサーバーとして利用する機能拡張が可能になる。

キーワード：論文データベース, PDF, 検索, Web

1 はじめに

インターネットが広く利用されるようになって、研究論文をキーワードでネットワーク上の論文アーカイブから引き出すことが日常的に行なわれている。これによって、新しい研究成果にしながらにして接することができるようになった。このような機能は Web サーバーをデータベースサーバとして、検索機能を付加することで実現されている。実験物理学の分野では特に日常的にこのような利用法が行なわれているが、今回はそれに類似したデータベースシステムを構築することができたので、その方法について報告する。データベースは、クライアントサーバーモデルを使った複雑なシステムとして運用されるが、ここではテキストベースの簡便なものを例としてとりあげる。ここで、Web ブラウザーから参照可能なものにするためには Common Gateway Interface [2] の機能を Web サーバー側で利用することが必要になる。また、データベースサーバーとして利用するとなると、アクセスに関する制限を設ける必要が生ずる場合がある。このための認証機能を付加する方法 [3] についても論ずる。またこのデータベースサーバーの機能を使って実際の授業に利用している例を紹介する。この他データベースサーバーのさまざまな利用法を提案する。

2 Common Gateway Interface の機能の Web サーバー側で利用方法

Web の機能を利用すると計算機の種類によることなく情報を受けとったり、情報をデータベースに蓄積したりできるようになる。さらに検索によって欲しい情報を瞬時に得ることが可能になる。計算機の専門家でなくてもこのようなことが簡単にできる機能を Web を使って利用するためには Common Gateway Interface という Web サーバーの機能を使う必要がある。

このような名前がついているのは Web サーバーの機能を果たすプログラム (httpd) が別のプログラムに検索の実行を頼むことになり、橋渡しの機能を演じるというところから来ている。ここでプログラムからプログラムのデータのやりとりがあるので入出力に関しての決まりを学んでおく必要がある。京都教育大学で学生の電子メールアドレスを検索する簡単なスクリプトを例にとって説明する。ここでは理学科の学生のデータが rigaku.txt というファイルになっているものとする。まず検索する名前を入力する欄を Web ページに表示する必要がある。そのための html ファイルは以下のようになる。

```
<html>
  <head>
    <title> rigaku mail address </title>
  </head>
  <body>
    <h2> 理学科メールアドレスの検索 </h2>
    名字は最初が大文字でヘボン式で入れて下さい。名前は小文字です。
    <form action = "/~keisan/cgi-bin/smail.pl" method = "GET">
      <input name = "names" size = 10> <br>
      <input type = "submit" value = "search">
      <input type = "reset" value = "reset">
    </form>
  </body>
</html>
```

ここで <form action = "/~keisan/cgi-bin/smail.pl" method = "GET"> という html タグで /~keisan/cgi-bin/smail.pl というプログラムを実行して検索を行なうことを httpd に対して指示している。<input name = "names" size = 10> というタグでは names という変数に入力データが入れられることを示している。<input type = "submit" value = "search"> というタグでは、search というボタンを押すと入力欄の情報が smail.pl というプログラムに渡されることを指定している。<input type = "reset" value = "reset"> というタグは入力欄のデータを消して、入れ直す時に使うボタンを表示する。

次に実際の検索を行なうプログラム smail.pl を以下に示す。これはパールというインタプリタを使うスクリプトプログラムになっており、日本語の検索ができるように jcode.pl というライブラリプログラムを使っている。jcode.pl は各々の cgi-bin に入れておく必要がある。

```
1  #!/usr/bin/perl
2  print "Content-type: text/html\n\n";
3
4  require 'jcode.pl';
5
```

```

6  if ( $ENV {'REQUEST_METHOD'} eq "POST" ) {
7      read (STDIN, $str, $ENV {'CONTENT_LENGTH'});
8  } else {
9      $str = $ENV {'QUERY_STRING'};
10 }
11
12 @part = split ('&', $str);
13 foreach $i (@part) {
14     ( $variable, $value ) = split ('=', $i);
15     $value = ~ tr/+// /;
16     $value = ~ s/%([0-9a-fA-F][0-9a-fA-F])/pack("c",hex($1))/eg;
17     &jcode' convert( *value, 'euc');
18     $value = ~ s/ </&lt;/g;
19     $value = ~ s/> /&gt;/g;
20     $value = ~ s/ \015\012/ \012/g;
21     $value = ~ s/ \015/ \012/g;
22     $cgi{ $variable } = $value;
23 }
24
25 $nkf = ~ /bin/nkf -j~;
26
27 $file = ~ rigaku. txt~;
28 $sname = $cgi {'names'};
29
30 open (FILE, ~ $file~) || die ~ can't open rigaku. txt.\n~;
31 $i = 0;
32 while ( <FILE> ){
33     if (/ $sname/){
34         ( $log, $pass, $uid, $gid, $wname, $home, $shell ) = split(/:/);
35         $found [ $i+1 ] = ~ <a href = \~ mailto: $log\@kyokyo-u.ac.jp\~>
36             $wname </a> <br> ~;
37         $i ++;
38     }
39 }
40 close (FILE);
41 # network likes JIS
42 open(BROWSE, ~ | $nkf~);

```

```

43 print BROWSE <<EOF;
44 <title> search result </title>
45 <h2> 検索の結果です </h2>
46 $i 人が検索されました。
47 <hr>
48 @found
49 <hr>
50 EOF
51
52 exit;

```

ここで行番号1から25迄は日本語で処理を行なうための一般的な方法となっている。17行めの `&jcode' convert (*value, 'euc')` という命令は計算機内部での日本語処理を UNIX で一般的な EUC というコードを使うということを示している。行番号25の `$nkf="/bin/nkf -j"` という命令は、ネットワークを通じて通信を行なう時には JIS コードを用いることが必要のところから、漢字コードを計算機内部で使っているコードから JIS コードに変換するために必要なものである。JIS コードは7ビットのコードなのでデータ通信を行なう時には必ず JIS コードにしておく。8ビットめを使うシフト JIS や EUC では通信の過程で文字化けがおきてしまう可能性がある。

行番号30から39迄で検索を実行している。28行めの `$sname = $cgi{'names'}` という命令で、検索したい人の名前が `sname` という変数に入れられている。

30行目でファイルを開き、FILE というポインタで参照される。32行目の `while ( <FILE> )` という命令で改行までの一行が `$_` という変数に読み込まれる。33行目の `if ( /$sname/ )` という命令で、`$_` と `sname` の一致する部分があるか否かが調べられる。

`rigaku.txt` はパスワードファイルの形式をしているのでそれぞれのデータが: で区切られている。`split(/:/)` という命令を使うとそれぞれのデータが分割されるので検索結果からすぐメールが打てるように、`<a href = "\mailto: $log\@kyokyo-u.ac.jp\"> $wname </a>` というタグに検索したログイン名 `$log` と氏名 `$wname` を入れている。

行番号41から50迄で検索結果をブラウザに表示させている。検索した氏名の部分はメールの発信を指示するタグになっているのですぐメールが打てるようになっている。

3 CGI プログラムのデバッグ

前節の入力欄を作る html タグで `<form action = "/~keisan/cgi-bin/smail.pl" method = "GET">` の中に `method = "GET"` という指定があるがこれは、CGI プログラムが実行される時 httpd からどのようにデータが渡されるかを示すものである。`method = "GET"` の時は環境変数 `QUERY_STRING` にデータが渡される。上の例で `smail.pl` をテストするには次のコマンドを実行する。

```
$ export QUERY_STRING="names=Abe"
$ ./smail.pl
```

perl のスクリプトを書き換える時は必ず、前もってこのようにテストしておくことを勧める。Web ブラウザから直接テストするとスクリプトが間違っていて、プログラムが実行されないときには document contains no data というメッセージが返って来るだけでどこが悪いのかわからない。document contains no data がでてきたときには Web ブラウザの Go to 欄に QUERY_STRING が表示されるのでデーターが長い場合はそれをコピーして環境変数 QUERY_STRING に代入し perl のスクリプトをテストするのが良いだろう。

method="POST" というデータの渡し方はデータが大きい場合に利用される。プログラムのデバッグの時にはこれを使う必要はない。

4 Web サーバーの論文データベースサーバとしての利用

データベースサーバといわれるものは一般に特別なプログラムによって書き込みや読み出しを行なうので、専門的な知識が必要になる。これに対して Web サーバーの CGI の機能を利用するものはあくまで簡単なものであるが使い勝手が良い。まず論文のデータベースを作ってみることを考えよう。データベースの区切りは改行を仮定している。研究室のデータベースとして利用するとする。データベース登録の入力用 html ファイルは以下のようになる。

```
<html>
<head> paper database input </head>
<body>
<h1> 研究室論文データベース登録 </h1>
<p> 高嶋研究室の人のみ、記入して下さい。
<form method="post" action="/~ryuichi/cgi-bin/paper_db.pl">
<p> 著者名 :
<input name="names">
<p> 論文の題名 :
<input name="title" size="30">
<p> キーワード :
<input name="keyw" size="40">
<p> アブストラクトの場所 : (for example /~ryuichi/takasima/kiyou97.txt)
<input name="abstl" size="40">
<p> ファイルの場所 : (for example /~ryuichi/takasima/kiyou97.pdf)
<input name="location" size="40">
<p> <input type="submit" value="送る">
<input type="reset" value="取り消し">
</form>
```

```
</body>
</html>
```

論文をネットワークから読めるようにするには、ポストスクリプト形式だとパソコンからは読めないで、PDF (Portable Document Format) 形式が良いと思われる。natsci には ghost script version 5.10J がインストールされているのでポストスクリプトファイルを以下のようにして変換する。

```
ps2pdf kiyou97.ps kiyou97.pdf
```

PDF ファイルは Acrobat Reader というプログラムが無料で配布されているので、それを使って表示したり、印刷することになる。パソコンのワープロからポストスクリプト形式を出力するにはポストスクリプトプリンターのプリンタードライバーを動かしておいて、ファイル出力を指定しておけば良い。それを natsci に転送してから PDF に変換する。

ここで /~ryuichi/takasima/kiyou97.ps という形で httpd に対して指定しておくということとは unix の計算機では ~ryuichi/public_html/takasima/kiyou97.pdf というのが実際のファイル名となる。

CGI スクリプトファイルは次のようになる。

```
# !/usr/bin/perl
# ここに smail.pl の 2 行目から 25 行目までの部分を入れる。
$db_1 = $cgi{'names'} ;
$db_2 = $cgi{'title'} ;
$db_3 = $cgi{'keyw'} ;
$db_4 = $cgi{'abstl'} ;
$db_5 = $cgi{'location'} ;

$output_file = "paper_db" ;

@outdata = ("Author : $db_1, ", "Keyword : $db_3, ",
            "Abstract : <a href = \" $db_4 \"> $db_2 </a> ",
            "Location : <a href = \" $db_5 \"> $db_2 </a> ");

open (FILE, ">> $output_file") || die "can't open paper_db\n" ;
print FILE "@outdata\n" ;
close (FILE) ;

# network likes JIS
```

```

open (BROWSE)
open (BROWSE, " | $nkf");
print BROWSE <<EOF;
<title> search result </title>
<h2> データベースに書き込みました </h2>
以下の通りに書き込みました
<hr>
@outdata <p>
<hr>
EOF

```

```
exit;
```

ここで最初の部分に `smail.pl` の 2 行目から 25 行目までの部分が入っていないなければならないが省略してある。

論文を PDF に変換しておくブラウザから見れて便利である。PDF をくっつけることも簡単である。しかし PDF をワープロに読み込むことは今の所できない。

5 検索プログラムの書き方

検索プログラムとしてもっとも簡単な例を示す。実用的なものとするためには著者やキーワードを別々に検索用にすると高速検索が可能になる。

```

#!/usr/bin/perl
# ここに smail.pl の 2 行目から 25 行目までの部分を入れる。
$db_1 = $cgi{'keyw'};

$input_file = "paper_db";

$i = 0;
open (FILE, "<$input_file") || die "can't open paper_db\n";
while (<FILE>) {
  if (/ $db_1/) {
    $found [$i+1] = "$_ <p> ";
    $i++;
  }
}
close (FILE);

# network likes JIS
open (BROWSE, " | $nkf");
print BROWSE <<EOF;

```

```
<title> search result </title>
```

```
<h2> 高嶋研究室論文データベース検索結果 <h2>
```

```
$i 件の論文が検索されました。
```

```
<hr>
```

```
@found
```

```
<hr>
```

```
EOF
```

```
exit ;
```

ここでは、データは改行で区分されている。すでにデータベースにファイルの参照のための html タグが埋め込んであるので検索した結果から、論文またはアブストラクトを参照できる。検索は著者名キーワード論文タイトルの中の単語から一致するものを探している。

6 データベースの書き換え

このテキスト形式データベースを書き換えるもっとも簡単な方法は、エディターを使って書き換えることである。パールスクリプトを使って書き換えるには、書き換えの箇所まで読みながら別のファイルに書き込んでいき、書き換えの部分を CGI を使って受けとって、書き込み残りの部分も書き込んで、最後にファイル名を変更するというやり方をする必要がある。このあたりが Postgres のようなデータベースサーバープログラムを使うことと比較して見劣りする部分である。

7 認証機能を付加する方法

研究論文の公表前のもので、グループの内部のみに公開したい場合にアクセスを制限する方法がある。あるディレクトリに .htaccess というファイルを作りその中に以下に示すような記述を入れておくパスワードによるアクセスの制限が可能になる。

この場合のパスワードファイルは ~takasima/bin/.private というものである。これにパスワードを登録するためには ~takasima/bin/htpasswd というプログラムを実行して登録を行なう必要がある。このプログラムのソースと makefile は natsci/~takasima のホームページにおいてある。

```
AuthUserFile/home/ryuichi/bin/.private
```

```
AuthGroupFile/dev/null
```

```
AuthName ByPassword
```

```
AuthType Basic
```

```
<limit GET>
```

```
require user ryuichi
```

```
require user hongo
```

```
</limit>
```

このような認証を行なうと .htaccess のおいてあるディレクトリー以下のディレクトリーもアクセスが制限される。また京都教育大学のみにアクセスを制限したい場合には以下の .htaccess ファイルを作成する。

```
<limit GET>
order deny, allow
deny from all
allow from kyokyo-u.ac.jp
</limit>
```

8 授業への Web サーバーの利用と利用事例の提案

Web の使い方を授業の課題としている例として、計算機実験をあげておく。この Uniform Resource Locator は <http://natsci.kyokyo-u.ac.jp/~keisan> である。この中の課題に、CGI を使ったプログラミングも採り入れている。実際には JAVA も Perl もと学生に習得を迫るのは酷なので、簡単な例題をデバッグするような作業をさせている。

CGI の機能を使ったデータベースサーバーを自分たちで使っていくと、以下に述べることに利用できる有用なデータベースを簡単に作ることができる。

- ◆ 簡単なニュースシステムのような使い方ができる。
- ◆ 教育に関する悩みごと相談室のようなプログラムを各こともできるし、事例に関してキーワードで検索できれば、さらに有用なものとなる。
- ◆ 教科教育の具体的な実験事例のデータベースを作ることも可能になるし、何よりも事例を日本中から集めることが簡単にできるようになる。
- ◆ 研究会の成果を簡単に蓄積できる。
- ◆ 事務的な処理に利用する。図書館の本の検索システムなども作ることができる。また備品管理なども行なうことができる。

テキスト形式のデータベースは複雑なプログラミングなしで利用できる点で使い良いものである。UNIX になれた人なら grep というコマンドが必要な文字列を含む行を捜し出してくれることを知っている。Perl ではこの機能を使っているだけである。ここのデータをカンマで区切っておけば Excel などの使いなれた表計算ソフトに書き込むこともできる。データベースとしてネットワークに表を載せる機能を利用できるとのことである。しかしこの場合は広くデータの記入を求めたり、データを検索したりすることはできない。

また自分で perl のスクリプトを書かなくても、wwbbs のような一般に使われているものを

持って来てもよい。これは以下の URL を参照する。<http://www.rescue.ne.jp/>

9 まとめ

Web を日常の研究活動に利用している人は多いが、その仕組みについてはあまり知られていないようである。特に Web をより双方向性を持った形にしていくやり方をまとめてみた。プログラミングはそれほど難しくないのでこのような利用法が学校での教育に利用されていくことを願っている。

また、論文をネットワーク上に載せることは、非常に有用である。雑誌の投稿においても Web に載せてあることを編集者に伝えれば良い。問題は著作権保護に関連して認証システムをきちんとすることにあるがそれもここに述べたやり方で解決できる。ここに述べたようなやり方で、理学科有志による「フォーラム理科教育」という研究交流誌を作成することになっている。

Web を使っていく上で注意すべきことは、サーバシステムの運用については比較的計算機に詳しい人が必要であるという点に注意しておくことである。この部分は業者に任せても良いであろう。パソコンユーザーが気軽にネットワークを利用して日常業務ができる時代は、もうすぐそこまで来ている。

Acknowledgement

理学科の学生の安部さんと久保さんには例題をつくってもらいました。

参考文献

- [1] 高嶋隆一ファイルサーバーシステムの構築 (1998), 京都教育大学教育実践年報 14号
- [2] 笹木他, ワンランク上のホームページのための HTML & CGI 入門 (1996), エーアイ出版
- [3] 鶴田誠, CGI ちょー入門 (1997) 広文社