

# 携帯電話による漢字検索システム

住谷 芳幸

文学部文化情報メディア学科文化メディア専攻

(2002年9月12日受理)

## Sino-Japanese Characters Retrieval for the Portable Telephone

Faculty of Humanities, Department of Humanities and Information,

Major in Cultural Studies and Information,

Gifu Women's University, 80 Taromaru, Gifu City, Japan (〒501 - 2592)

SUMIYA Yoshiyuki

(Received September 12, 2002)

### 1 はじめに

携帯電話(以下、単に携帯とする。)の普及には目を見張るものがある。NTT ドコモは、2002年2月15日に携帯加入台数が四千万台を突破したと発表した。内、約三千百万台がi-MODE利用者であるという。翌日2月16日の朝日新聞の記事では、「メール送受信やニュース・娯楽情報の入手といった「ケータイからネット」は国民にはほぼ定着した形だ。」と評価している。確かに、文字の入力さえ問題にしなければ、携帯電話からのメール・インターネット接続は便利である。特にMS-Windowsの起動の遅さにくらべれば、携帯は軽快である。ところで、東京大学の坂村健教授は、早くから「どこでもコンピュータ」という提唱をしているが、携帯の普及は機能的に劣るもののそれを実現していると言えるのではなからうか。問題は、コンテンツの充実であろう。前記の朝日新聞の記事でも、「メール送受信やニュース・娯楽情報の入手」とあるのみで、教育・研究についての言及はない。教育・研究を目的として、携帯をどの様に利用するか、あるいはどの様に利用できるかは、今後の検討課題と言えよう。

そこで、携帯から利用できる漢字検索システムを計画してみた。以前に広韻索引を公開したが、そのデータを用いて携帯から特定の漢字について広韻での掲出頁等を知るためのシステムである。例えば、演習中に漢字1字につき、その広韻での所在を調べるために、いちいちコンピュータを起動するのは、あまり実用的とは思われない。コンピュータが起動するまでの時間で、紙の索引が引けてしまうであろう。携帯のbookmarkに携帯用の広韻検索システムが登録されていれば、すぐさま検索が可能であろう。また、寺院等で調査の際に電源が取れずにコンピュータが使えないこともある。そのような際に、携帯から広韻の検索ができれば便利であろう。

ところで、一般にインターネット上で検索システムを構築する際には、Perlなどによるcgiプログラムが利用される。ただし、cgiプログラムは素人には敷居が高いであろう。また、シ

システムによっては cgi を禁止している場合もある。そのため、cgi 以外に何かよい方法がないか検討している時、管理工学研究所が kiricgi.exe というフリーソフトを公開していることに気付いた。kiricgi.exe は、管理工学研究所が販売しているデータベースソフト桐 Ver 8 の一括処理により cgi を実現させるためのプログラムである。また、ikiri.cmd という一括処理のサンプルプログラムも添付されていた。検討の結果、これを利用すれば簡単に携帯用の漢字システムを構築できそうである。桐の一括処理は始めてではあるがなんとかなるであろうと考え、データ整理・一括処理の記述等を始め、なんとか外部向けに公開できた。ここでは、その過程及びその評価につき書いてみたい。

## 2 システムの概要

システムは概要は次のようになる。

### 1) 携帯用 WEB ページ (i 広韻)

これは index.htm・setumei.htm・kensaku.htm から構成される。index.htm は i 広韻のトップにあたり、説明画面・検索画面にリンクが張ってある。setumei.htm は i 広韻を説明するための説明画面のページである。kensaku.htm は kiricgi.exe に検索用のデータを送るための検索画面のページである。データは HTML の FORM により送信される。

### 2) kiricgi.exe

上記の kensaku.htm よりデータを受け取った kiricgi.exe は、そのデータを特定のディレクトリに req.txt として書き出す。また、桐の一括処理により reply.html として書き出された検索結果を携帯に送り返す。

### 3) ikoin.cmd (桐・一括処理)

kiricgi.exe が書き出した req.txt を読み込み、携帯から要求された検索を行い、その結果を reply.html として書き出す。

これだけの事で、携帯から漢字の検索が可能になる。以下ではシステムを具体的に説明する。

## 3 システムの具体的な説明

### 3.1 携帯用 WEB ページ (i 広韻)

ここでは検索の中心である kensaku.htm について説明する。kensaku.htm は次の様になっている。なお、説明のために行番号を付加した。

```
1 : HTML
2 : HEAD
3 : META HTTP-EQUIV="content-type" CONTENT="text/html;charset=Shift_JIS"
4 : META NAME="description" CONTENT="iMODE 広韻検索画面"
5 : META NAME="keywords" CONTENT="iMODE, 広韻検索, 韻鏡検索"
6 : TITLE i 広韻検索画面 /TITLE
7 : /HEAD
```

```

8 : BODY BGCOLOR="#ffff00" TEXT="#000000"
9 : P
10 : CENTER FONT COLOR="#ff0000"   &#63861 ; 広韻   /FONT /CENTER
11 : HR SIZE="4"
12 : CENTER  検索画面  /CENTER
13 : BR
14 : FORM METHOD="get" ACTION="scripts/kiricgi.exe"
15 : INPUT TYPE="radio" NAME="検索" VALUE="広韻" CHECKED  広韻検索  BR
16 : INPUT TYPE="radio" NAME="検索" VALUE="韻鏡"  韻鏡検索  BR
17 : 漢字: INPUT TYPE="text" NAME="漢字" SIZE="2" MAXLENGTH="2" BR
18 : INPUT TYPE="submit" VALUE="送信"
19 : /FORM
20 : HR SIZE="4"
21 : /P /BODY
22 : /HTML

```

検索画面の中心部分は14行から19行目である。

15行・16行では、ラジオボタンにより検索の種別、「広韻検索」か「韻鏡検索」かのいずれかを選択するようにしてある。(初期値は「広韻検索」。)送信ボタンがおされるとNAMEとしては「検索」、VALUEとしては「広韻検索」であれば「広韻」が、「韻鏡検索」であれば「韻鏡」が送信される。

17行目のテキストボックス漢字入力の項目では、MAXLENGTH="2"と漢字1字しか入力できないようにしてある。送信ボタンがおされるとNAMEとしては「漢字」、VALUEとしてはテキストボックスで入力された漢字が送信される。

18行目は検索ボタンとなる。検索ボタンがおされると、14行目のHTMLのFORMによりディレクトリ scripts にある kiricgi.exe にリクエストメソッド get で上記のデータを送信する。なお、kiricgi.exe はリクエストメソッドとして、post と get に対応しているが、ここでは get を用いている。名前を「i 広韻」としているが、i-MODE 用の絵文字(10行目の&#63861;)を使っているだけで、特殊なHTMLで書かれているわけではない。そのため、J-Sky 対応の携帯でも見えるはずである。ただし、一部のJ-Sky 対応の携帯ではリクエストメソッド post に対応していない。(私の使っているJ-Sky 携帯では post を含む行を全く表示しなかった。)そのためリクエストメソッド get でデータを送信することにした。また「i 広韻」の場合、送信する漢字は1文字であるため、リクエストメソッドとして get を用いても特に問題はない。

### 3.2 kiricgi.exe

kensaku.htm よりデータを受け取った kiricgi.exe が書き出す req.txt はつぎの様である。検索種別を「広韻検索」、検索すべき漢字を「亜」として送信した時のものである。なお、都合により省略した部分がある。

```

"ENV","BLASTER","A220 I5 D1 T4"
"ENV","CMDLINE","WIN"
"ENV","COMSPEC","C:\WINDOWS\COMMAND.COM"
"ENV","CONTENT_LENGTH","0"
"ENV","DMISVAPATH","C:\sva\dmibin"
"ENV","GATEWAY_INTERFACE","CGI/1.1"
"ENV","HTTP_USER_AGENT","DoCoMo/1.0/N502i"
"ENV","IPF_PATH32","c:\IPFWIN"
"ENV","PATH","/WINDOWS;/WINDOWS/COMMAND;/SVA/DMI/BIN;/WNETFIN"
"ENV","PROMPT","$p$g"
"ENV","QUERY_STRING","%8C%9F%8D%F5=%8D%4C%89%43&%8A%BF%8E%9A=%88%9F"
"ENV","REMOTE_ADDR","210.136.161.72"
"ENV","REMOTE_HOST","210.136.161.72"
"ENV","REQUEST_METHOD","GET"
"ENV","SERVER_PORT","80"
"ENV","SERVER_PORT_SECURE","0"
"ENV","SERVER_PROTOCOL","HTTP/1.0"
"ENV","TEMP","C:\WINDOWS\TEMP"
"ENV","TMP","C:\WINDOWS\TEMP"
"ENV","WINBOOTDIR","C:\WINDOWS"
"ENV","WINDIR","C:\WINDOWS"
"ENV","TERM","cygwin"
"GET","検索","広韻"
"GET","漢字","亜"

```

なお、次の一括処理で必要なのは最後に書き込まれた2行のみである。ここに検索の種類(「広韻」と検索すべき漢字(「亜」と)が書き込まれているからである。

### 3.3 i-koin.cmd (桐・一括処理)

桐の一括処理は次の様になっている。なお、説明のために行番号を付加した。

```

1 : * ----- *
2 : *           携帯用 広韻・韻鏡検索システム           *
3 : * ----- *
4 :
5 : 定数宣言 固有, 文字列 ¥
6 :     { &TempDir      = "c:\k3\data\kiricgi" , ¥
7 :     &KoinTable     = "c:\k3\data\koin.tbl" , ¥

```

```

8 :          &PermitTable      = "c:¥k3¥data¥cgi.tbl"    }
9 : 定数宣言  固有, 文字列 ¥
10 :         { &RequestFile     = &TempDir + "req.txt" ,¥
11 :           &ReplyFile       = &TempDir + "reply.html" ,¥
12 :           &ReplyWorkFile    = &TempDir + "reply.txt" ,¥
13 :           &method          = "GET"          }
14 : 変数宣言  固有, 整数 { &break , &stat }
15 : 変数宣言  固有, 文字列 { &value , &kanji }
16 :
17 : 表 &KoinTable
18 : 表 &PermitTable
19 :   行削除 * , 圧縮
20 :
21 : 繰り返し
22 :   &stat = 0
23 :   繰り返し ( &stat  1 )
24 :   条件 ( &break = 27 )  中止
25 :   遅延  1
26 :   機能キー入力  ( 12 , 25 ) , プロンプト = " Esc キーで終了します " , ¥
27 :               入力待ち      = しない , ¥
28 :               バッファクリア = しない , ¥
29 :               &break
30 :   ファイル入力開始 &RequestFile , 終了状態 = &stat
31 :   繰り返し終了
32 :   ファイル入力終了
33 :
34 :   * req.txt を作業表に読み込む。
35 :   編集表 &PermitTable
36 :   読み込み テキスト , &RequestFile , 区切り = " ; " , *
37 :   * 作業表の項目は [type],[name],[value]。
38 :   * 項目 [type] の値が GET の場合だけ取り出す。
39 :   絞り込み [type]{ &method }
40 :   * 検索する漢字を &kanji に代入。用心のために先頭だけ取り出す。
41 :   絞り込み [name]{ "漢字" }
42 :   &kanji = #部分列 ( [value] , 1 , 1 )
43 :   * &value に検索の種類 ( 広韻か韻鏡か ) を代入。
44 :   絞り込み 補集合
45 :   &value = [value]

```

46 :  
47 : ケース開始  
48 : \* 漢字が送られているかどうかを確認。  
49 : ケース ( #IS 漢字 ( &kanji , 1 ) = 0 )  
50 : 手続き実行漢字確認  
51 : \* 検索種別により検索結果を書き出す。  
52 : ケース ( &value = "広韻" )  
53 : 手続き実行広韻書出  
54 : ケース ( &value = "韻鏡" )  
55 : 手続き実行韻鏡書出  
56 : ケース終了  
57 : \* 作業表の後始末  
58 : 編集表 &PermitTable  
59 : 絞り込み解除 \*  
60 : 行削除 \* , 圧縮  
61 : 繰り返し終了  
62 : 終了 表 \*  
63 : 終了  
64 :  
65 : 名札 漢字確認  
66 : 印字開始&ReplyWorkFile  
67 : 印字" HTML "  
68 : 印字" HEAD "  
69 : 印字" META HTTP-EQUIV=""content-type"" CONTENT=""text/html;charset=Shift\_JIS"" "  
70 : 印字" /HEAD "  
71 : 印字" BODY BGCOLOR=""#ffff00""TEXT=""#000000"" "  
72 : 印字" CENTER FONT COLOR=""#ff0000"" &#63861広韻 /FONT /CENTER "  
73 : 印字" HR SIZE=""4"" "  
74 : 印字" 検索結果 BR BR "  
75 : 印字"漢字を送信してください。 BR BR "  
76 : 印字" FORM METHOD=""get"" ACTION=""kensaku.htm"" "  
77 : 印字" INPUT TYPE=""submit"" VALUE=""検索画面へ"" "  
78 : 印字" /FORM "  
79 : 印字" HR SIZE=""4"" "  
80 : 印字" /BODY "  
81 : 印字" /HTML "

82 : 印字終了

83 : 絞り込み解除 \*

84 : ファイル名変更&ReplyWorkFile , &ReplyFile

85 : ファイル削除&RequestFile

86 : &value = "" , &kanji = ""

87 : 手続き終了

88 :

89 : 名札 広韻書出

90 : 編集表 &KoinTable

91 : 絞り込み [ 漢字 ] { &kanji }

92 : 印字開始&ReplyWorkFile

93 : 印字” HTML ”

94 : 印字” HEAD ”

95 : 印字” META HTTP-EQUIV=""content-type"" CONTENT=""text/html;charset=Shift\_JIS"" ”

96 : 印字” /HEAD ”

97 : 印字” BODY BGCOLOR=""#ffff00""TEXT=""#000000"" ”

98 : 印字” CENTER FONT COLOR=""#ff0000"" &#63861広韻 /FONT /CENTER ”

99 : 印字” HR SIZE=""4"" ”

100 : 印字” [ "+&kanji+" ]の検索結果 BR BR ”

101 : 繰り返し ( . not#終端行 )

102 : 印字 [ 広韻表示 ]

103 : 印字 [ 備考 ]

104 : ジャンプ 行番号 = 次行

105 : 繰り返し終了

106 : 印字” BR ”

107 : 印字” FORM METHOD=""get"" ACTION=""kensaku.htm"" ”

108 : 印字” INPUT TYPE=""submit"" VALUE=""検索画面へ"" ”

109 : 印字” /FORM ”

110 : 印字” HR SIZE=""4"" ”

111 : 印字” /BODY ”

112 : 印字” /HTML ”

113 : 印字終了

114 : 絞り込み解除 \*

115 : ファイル名変更&ReplyWorkFile , &ReplyFile

116 : ファイル削除&RequestFile

117 : &value = "" , &kanji = ""

```

118 : 手続き終了
119 :
120 : 名札 韻鏡書出
121 :     編集表 &KoinTable
122 :     絞り込み [ 漢字 ]{ &kanji }
123 :     印字開始&ReplyWorkFile
124 :     印字” HTML ”
125 :     印字” HEAD ”
126 :     印字” META HTTP-EQUIV=””content-type”” CONTENT=””text/html;charset=
Shift_JIS”” ”
127 :     印字” /HEAD ”
128 :     印字” BODY BGCOLOR=””#ffff00””TEXT=””#000000”” ”
129 :     印字” CENTER FONT COLOR=””#ff0000”” &#63861広韻 /FONT /
CENTER ”
130 :     印字” HR SIZE=””4”” ”
131 :     印字”[ ”+&kanji+” ]の検索結果 br br ”
132 :     繰り返し (.not#終端行)
133 :     印字 [ 韻鏡表示 ]
134 :     印字 [ 備考 ]
135 :     ジャンプ 行番号 = 次行
136 :     繰り返し終了
137 :     印字” BR ”
138 :     印字” FORM METHOD=””get”” ACTION=””kensaku.htm”” ”
139 :     印字” INPUT TYPE=””submit”” VALUE=””検索画面へ”” ”
140 :     印字” /FORM ”
141 :     印字” HR SIZE=””4”” ”
142 :     印字” /BODY ”
143 :     印字” /HTML ”
144 :     印字終了
145 :     絞り込み解除 *
146 :     ファイル名変更&ReplyWorkFile , &ReplyFile
147 :     ファイル削除&RequestFile
148 :     &value = ”” , &kanji =””
149 : 手続き終了

```

5行目から32目までは前処理の部分である。なお、この部分は kiricgi.exe 添付のサンプルプログラムを、ほぼそのまま踏襲している。定数&KoinTableとして ikoin.tblを使用し、また変数&kanjiを追加した程度しか変更部分はない。



36行目で kiricgi.exe が書き出した req.txt を cgi.tbl に読み込む。cgi.tbl は method,name,value の3つの項目からなる桐の表である。39行目で、cgi.tbl の method の項目が「GET」のものだけを選択する。前記 req.txt の最後の2行が選択されることになる。さらに name の項目が「漢字」のものに絞り込む。絞り込まれたレコードの value の項目に携帯から送られた検索すべき漢字が含まれるため、その値を変数&kanji に代入しておく。なお、kensaku.htm では漢字1文字のみが送信されるようにしてあるのだが、用心のため#部分列([value], 1, 1)として、先頭の1字だけを取り出しておく。さらに補集合を選択すると、name の項目値が「検索」のレコードとなる。そのレコードの、value の項目に検索の種別、「広韻」か「韻鏡」かのいずれかが含まれるため、その値を変数&value に代入しておく。

以上の下処理が終わるとメインの処理に移る。ケースを使うことにより変数&kanji・&value の値によりその処理を変えることになる。

49行では、#IS 漢字(&kanji, 1)の値を確認することで、変数&kanji に代入されたデータが、漢字であるかどうかの確認をしている。#IS 漢字(&kanji, 1)の値が0の場合、その&kanji に含まれる文字が漢字以外であるため65行目の名札：漢字確認以下に移り、印字コマンドにより以下の確認用の reply.html を書き出す。

```
HTML
HEAD
META HTTP-EQUIV="content-type" CONTENT="text/html;charset=Shift_JIS"
/HEAD
BODY BGCOLOR="#ffff00"TEXT="#000000"
CENTER FONT COLOR="#ff0000" &#63861広韻 /FONT /CENTER
HR SIZE="4"
 検索結果 BR BR
漢字を送信してください。 BR BR
FORM METHOD="get" ACTION="kensaku.htm"
INPUT TYPE="submit" VALUE="検索画面へ"
/FORM
HR SIZE="4"
/BODY
/HTML
```

これが kiricgi.exe により送信元の携帯に送り返される。

さらに68行目で変数&kanji・&value の値はクリアされるため、52行目以下の処理は行われな  
い。さらに、60行目で cgi.tbl の全てのデータを削除し、22行目から31行目目までの入力待ちに  
戻ることになる。

#IS 漢字(&kanji, 1)の値が1の場合は52行目以下の処理に移り、ikoin.tbl を使用して目的  
の漢字の検索を行う。ここで用いている ikoin.tbl は漢字、広韻表示、韻鏡表示、備考の項目か

らなる桐の表である。漢字の項目には ShiftJIS に含まれる6355字の漢字が全て含まれている。なお、広韻に於いて重複して掲出されている漢字があるため、全体では8473レコードとなっている。広韻表示の項目には ikoin.tbl の漢字の項目に含まれた漢字についての、広韻での頁・行数・行内での順番が、

「421頁 8行第7字[示馬]韻) BR 」

の様な形式で入っている。(漢字「亜」の項目。)なお、ここでの頁数等は藝文印書館『校正宋本廣韻』によった。また、ShiftJIS に含まれない漢字は [示馬] 韻の [示馬] の様に漢字2字を組合わせて表示した。韻鏡表示の項目には ikoin.tbl の漢字の項目に含まれた漢字についての、韻鏡での転、五音、清濁、声調、等位、三十六字母が、

「第29転、喉音、清、去声、2等、影母 BR 」

の様な形式で入っている。(漢字「亜」の項目。)備考には異体字の情報等が、

「[広韻、亞] BR 」

の様な形式で入っている。(漢字「亜」の項目。)この場合、広韻では検索した漢字「亜」ではなく「亞」として掲出されていることを示す。また、検索した漢字が広韻に含まれない場合は、広韻表示・韻鏡表示には

「この字、広韻に無し BR 」

というデータが入れてある。さらに、備考には他の韻書にその字が掲出されている場合にはその反切等の情報が入れてある。ところで、これらの項目末の「 BR 」は HTML での改行用のタグである。改行用のタグを一括処理の印字コマンドで書き出すことも可能ではあるが、ここでは ikoin.tbl のデータ内に含めた。なお、以前に公開した広韻索引では頁・行数・行内での順番等は別項目としたが、今回は書き出し用として1項目にまとめた。

さて、変数&value の値が「広韻」の場合は53行目で名札：広韻書出に処理を移す。変数&value の値が「韻鏡」の場合は55行目で名札：韻鏡書出に処理を移す。ともに ikoin.tbl の漢字の項目を、変数&kanji の値で選択をする。(91行目・122行目)そして広韻書出では ikoin.tbl の項目、広韻表示と備考が、印字コマンドにより以下の reply.html として書き出される。これが広韻検索の結果となる。

HTML

HEAD

META HTTP-EQUIV="content-type" CONTENT="text/html;charset=Shift\_JIS"

```

/HEAD
BODY BGCOLOR="#ffff00"TEXT="#000000"
CENTER FONT COLOR="#ff0000"   &#63861広韻   /FONT /CENTER
HR SIZE="4"
[ 亜 ] の検索結果 BR BR
421頁 8 行第 7 字([ 示馬 ] 韻) BR
[ 広韻, 亞 ] BR
BR
FORM METHOD="get" ACTION="kensaku.htm"
INPUT TYPE="submit" VALUE="検索画面へ"
/FORM
HR SIZE="4"
/BODY
/HTML

```

韻鏡書出では ikoin.tbl の項目、韻鏡表示と備考が、印字コマンドにより以下の reply.html として書き出される。これが韻鏡検索の結果となる。

```

HTML
HEAD
META HTTP-EQUIV="content-type" CONTENT="text/html;charset=Shift_JIS"
/HEAD
BODY BGCOLOR="#ffff00"TEXT="#000000"
CENTER FONT COLOR="#ff0000"   &#63861広韻   /FONT /CENTER
HR SIZE="4"
[ 亜 ] の検索結果 BR BR
第29転, 喉音, 清, 去声, 2 等, 影母 BR
[ 広韻, 亞 ] BR
BR
FORM METHOD="get" ACTION="kensaku.htm"
INPUT TYPE="submit" VALUE="検索画面へ"
/FORM
HR SIZE="4"
/BODY
/HTML

```

これらの処理を終えると ikoin.tbl の絞り込みを解除し,変数&kanji・&value をクリアする。( 114 行以下・145行以下)

最後に、cgi.tbl の項目をすべて削除して、入力待ちの状態に移る。(58行以下)

ところで、既に書いた様に以上の一括処理の大部分はサンプルプログラムからの流用である。これは、あながち、私の不勉強あるいは怠慢からというばかりではない。kiricgi.exe が req.txt を書き出す、reply.html を送り返すという働きしかしないため、他に書きようが無いというのが実情である。

#### 4 まとめ

以上をインターネット上に公開し、簡単なものではあるが携帯による漢字検索が実現している。今回のシステムではデータの制約から、広韻検索・韻鏡検索の2つの検索のみであるが、データさえあれば、さらに複雑・高度な検索システムも可能であろう。また、ここでは kiricgi.exe を用いた携帯用の漢字検索システムについて説明してきたが、kiricgi.exe 自体は携帯用に特化したプログラムではない。あくまでも、リクエストメソッド post・get で送られたデータを req.txt として書き出し、送信元に reply.html を送り返すプログラムである。そのため、携帯からではなく、コンピュータからの送信に対しても対応可能であり、簡単に cgi を利用したい時には便利であろうし、また、桐による複雑な検索も可能である。ただし、問題がないわけではない。

その一つは、kiricgi.exe が req.txt を書き出すディレクトリが固定されていることである。req.txt を書き出すディレクトリが変更できれば、例えば、広韻検索である ikoin.htm と源氏物語検索である igenji.htm を用意し、桐を二重に起動してそれぞれの処理をさせることも可能であろう。しかし、現状ではそのようなことは不可能である。もっとも、kensaku.htm でラジオボタンにより、広韻検索か韻鏡検索かのいずれかを選択したように、広韻検索と源氏物語検索とのいずれかを選択して検索することは可能である。問題は、広韻検索と源氏物語検索とのような内容的に異質のものを並べて選択させることに、感覚・感情的に問題がないかどうかだけである。

また、桐が扱える文字コードは現在のところ ShiftJIS のみであり、UNICODE には対応していない。そのため、研究用の漢字データ作成には文字数の上で問題がないわけではない。ただし、携帯自体は ShiftJIS 対応であるため、今回のデータとしては問題はない。

携帯で漢字を入力する場合は、仮名漢字変換により目的の漢字を探すことになる。それでもない場合は、私の持つ i-MODE 用の携帯では区点入力が可能である。ただし、区点入力は機種により違いがあるようである。私の持つ J-Sky 用の携帯には区点入力はないようである。もっとも区点入力があっても、区点コードを知るためには、コード字典なりコード表を持ち歩く必要がある。このことは携帯の簡便さ、あるいは携帯からの漢字検索の簡便さには反すると思われる。携帯による簡便さを求めるならば、漢字の部首、画数からも目的の漢字を検索できるシステムを考えるべきであろう。今後の課題としたい。

今回は、広韻検索という検索されるデータ量がそれほど多くならない事が予想されるデータを扱った。ただし、携帯による本格的な検索システムを考える場合、検索するデータ量が多い場合も想定される。その際には、検索データを分割して表示する事は可能ではあろう。ただし、

その場合、送受信を繰り返すことになり、携帯による検索という簡便に反することになると思われる。携帯による検索システムは、今回の広韻検索の様な検索されるデータ量がそれほど多くなることが予想されるものにむくのではあるまいか。

さて、筆者もときどきこの広韻検索を使用するが、検索結果を紙に書き出すことがある。現在でも様々なデータが i-MODE 用に公開されており、何度も同じデータにアクセスすることも多い。その様な際に、プリンタ付きの携帯があれば考えるのは筆者だけであろうか。

最後に、実際に広韻検索システムは2002年5月より稼働中です。もし必要であれば、

<http://www.gijodai.ac.jp/i/ikoin/index.htm>

に接続してください。ただし、システムの都合により予告なく閉鎖することもあります。