パソコン用地学かな漢字対応表の作成とワープロ辞書としての利用について

野呂春文・村田泰章・佐藤岱生・小松崎峰子（地質情報解析室）
Harufumi Noro・Yasuaki Murata・Taisei Sato・Mineko Komatsuzaki

はじめに

パソコンワープロや 日本語フロントエンドプロセッサーを用いたプログラムでの入力作業の能率は “かな” 漢字変換辞書がどれだけ充実しているかで決まるといっても過言ではありません. 購入したばかりのワープロソフトについている辞書では まったく不十分で 必要な単語を追加して はじめて 使いものになるのがふつうです. そのため パソコンワープロの上手な人とは 辞書登録をどんどん気楽にやる人のことだと言われるほどです.

とはいえ 辞書登録を系統的におこなうことは 大変めんどうで退屈な作業です. “かな” 漢字変換辞書の充実が大事であることはわかっても 個人では なかなか十分なものが作れないのが実状でしょう.

地質情報解析室では すでに工業技術院共同利用研究情報システム（RIPS）の FACOM メインフレーム用に “かな” 漢字変換辞書を作成していましたので（佐藤ほか 1987） その成果をもとにして パソコンワープロの辞書を充実させるための作業に着手し このたび一応の完成をみました. ここに その作業過程使用法等を公開して みなさんの参考に供したいとおもいます.

```
あいのうらがた，相浦型，1
あいのうらそう，相浦層，1
あいのさわそう，合ノ沢層，1
あいのしまたい，相ノ島帯，1
あいのしまたい，相島帯，1
あえん，亜円，1
あえんがんたい，亜沿岸帯，1
あえんはく，亜鉛白，1
あおいし，青石，1
あおがしま，青ガ島，1
```

図1 パソコン用地学かな漢字対応表のファイル形式.
一行が 一つの単語の よみがな 漢字 属性を表現している. 属性の項は「一太郎」特有のもので数字1は 普通名詞を意味している. 他のワープロソフト 他のコンピュータで 地学かな漢字対応表を使う場合は ファイル形式の変更が 必要である.

1. どのようなものを作ったか

作ったのは 図1のような形式の MS-DOS のテクストファイルです. 一行が 一つの単語の “かな” と漢字の対応を表しています. 各行は まず 単語の読みのひらがな 全角のコンマ 漢字表記 全角のコンマ そして 全角の数字からできています.

このような形式にしたのは 実は ワープロソフト「一太郎 Ver. 3」の一括辞書登録機能を利用するためなのです. 一太郎 Ver. 3 では 図1のような形式のファイルを用意しておけば 7千語までの単語が 一度に辞書登録できます. ただし 後でも述べますが この “かな” 漢字対応表は わずかな変異で 他のワープロソフトに対応できます. 決して「一太郎」専用ではありません.

“かな” 漢字変換辞書の形式はワープロソフトごとにまったく異なっています. そのため すべてのワープロソフトごとの辞書を作るようなことは 一研究室の能力を越えています. しかも 特定のワープロソフト用の辞書を作った場合 そのコピーを他の人にあげることが 使用許諾権契約で禁止されていることが多いのです. せっかく辞書を作っても他の人が使えないのでは意味がありません. そこで 上のような 単純な “かな” と漢字の対応表を作成したわけです. 各自 この対応表を 現在使っているワープロソフトの辞書に組み込んで 辞書を作成していただく という考えです.

この “かな” 漢字対応表の単語の総数は 約2万8千語です. しかし 上で述べたように 一太郎の一括辞書登録機能を利用する場合は 7千語が限度ですので 一太郎用に 単語総数を7千語におさえた “かな” 漢字対応表も作成しました. それは (1) 地学用語（chigaku. dic） 5,683単語 (2) 地名（chimei. dic） 4,141単語 (3) 国土地理院発行図葉名（zumei. dic） 1,858単語 (4) 山川等の自然地名（yamakawa. dic） 5,689単語 の4つのファイルです.

この “かな” 漢字対応表に収録している単語の範囲に

```
／あいのうらがた／相浦型／
／あいのうらそう／相浦層／
／あいのさわそう／合ノ沢層／
／あいのしまたい／相ノ島帯／
／あいのしまたい／相島帯／
／あえん／亜円／
／あえんがんたい／亜沿岸帯／
／あえんはく／亜鉛白／
／あおいし／青石／
／あおがしま／青ガ島／
```

図2　東芝 AS3000 シリーズ unix ワークステーションで 辞書登録を行う場合の“かな”漢字対応表の形式. 一行が '／' で始まりよみがな '／' 漢字 そして '／' で終わる. '／' は 半角でも全角でも良い.

ついては 佐藤ら (1987) にくわしく述べられていますので ここでは 簡単に説明するにとどめます.

まず 地名です. 地名には 自然地名と行政地名とがあります. 行政地名は 大方のワープロの辞書では村のレベルまで収録されていることが多いので この“かな”漢字対応表では 山 川 湖 岬などの いわゆる自然地名を主にして収録しています. また 国土地理院発行の地形図の図葉名も収録しています.

地球科学用語は 事典の見出し語から選択しています. よほど特殊な単語は別にして 国内発行の学会誌や書物にあらわれる単語のほとんどを網羅していると思います.

2. ワープロ辞書として利用する

パソコンワープロの辞書として“かな”漢字対応表を使う方法の一例を紹介します. ここでは 一太郎の例しか紹介できませんが 他のワープロソフトでも同じようなことができるかもしれません. 使っているワープロソフトのマニュアルをよく見てください.

一太郎の場合 一つの辞書に1で紹介した4つの“かな”漢字対応表全部を登録することはできませんので 必要に応じて ひとつか あるいは 総数7千語を越えない範囲でいくつかを選んで 辞書登録をおこなうことになります.

辞書登録の手順は 以下のとおりです.

(1) 初期化の済んだフロッピーディスクを2枚用意します.

(2) その一枚に 現在使用中の辞書 (ATOK. DIC) と一太郎用“かな”漢字対応表 (chigaku. dic その他) をコピーします.

(3) もう一枚のフロッピーディスクは 新辞書用ディスクとして あとで使いますのでそのままにしておきます.

(4) Aドライブに一太郎についてくる「ユーティリティーディスク」をセットして UT (または ut) と入力します. あとは 指示にしたがって作業を進めれば新しい辞書ができあがります. 地学用語 (chigaku. dic) 約5千7百単語を登録した場合 だいたい1時間かかります.

(5) できあがった新しい辞書を 現在使用しているフロッピーディスクまたは ハードディスクにコピーして作業終了です.

繰り返しになりますが この“かな”漢字対応表の利用は 一太郎のみに限られるわけではありません. 「新松」等 他のパソコンワープロソフト 多くの unix ワークステーションでは 一括辞書登録機能を持っていますので その仕様に合わせて“かな”漢字対応表の形式を変更すれば それらの計算機で利用できるでしょう.

東芝 AS3000 シリーズワークステーションの場合漢字コードを拡張 unix コード (後述) に変換して図2のような形式の“かな”漢字対応表を用意すれば 簡単に辞書の拡張ができます.

SONY の NEWS 京都大学数理解析研究所から公開されている GMW (萩谷 1987) のフロントエンドである Wnn (桜川 1987) などでも 多少変更すれば この“かな”漢字対応表が使えると思います.

“かな”漢字対応表は MS-DOS のテクストファイルですから 通常のテクストエディターで 単語 “読み”の 修正 追加 削除ができます. ワープロソフトでも この作業ができないことはありませんが ファイルが大変大きいので エディターを使ったほうが楽でしょう.

“かな”漢字対応表を 他のワープロソフトや システムに対応させるには 仕様に合わせた 簡単なフィルターに通すだけでできます. フィルターとは ファイルを読んで なんらかの加工を施してから ふたたびファイルに書き出す仕事をするプログラムのことです.

一例として 付録に JEF 漢字コードをシフト JIS 漢字コードに変換するためのフィルタープログラムを載せておきますので参考にしてください.

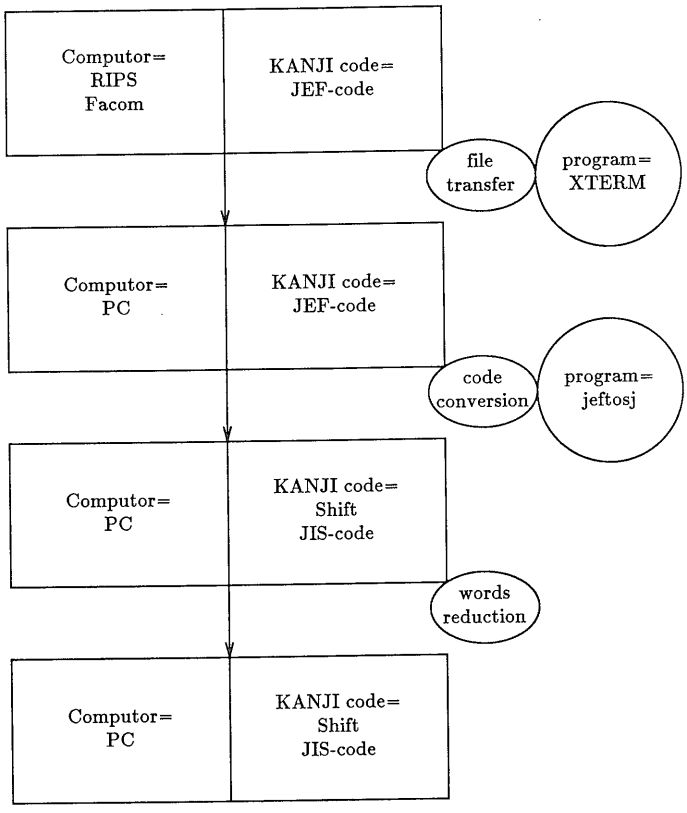

図3　パソコン用地学かな漢字対応表の作成過程．四角の箱の中は　ファイルの所在と漢字コードを示す．楕円の中は作業を　円の中は　そのために用いられたプログラム名を　それぞれ示す．

作業は　それぞれ　(1)　RIPS 上の JEF コードのファイルを PC に転送する　(2)　PC 上で JEF コードからシフト JIS コードに変換する　(3)　PC 上で　単語を削除する　である．

3.　どのようにして作ったか

“かな”漢字対応表を　どのようにして作ったか　簡単に説明しましょう (図3)．

工業技術院筑波研究センターの共同利用計算機 (RIPS) の FACOM　M380 で利用できる形の“かな”漢字変換辞書が　情報解析室の佐藤らによって　約2年がかりで作られました (佐藤ほか　1987)．

この“かな”漢字変換辞書 (以降　これをオリジナル辞書と呼びます) が　今回作成した“かな”漢字対応表の母体です．　オリジナル辞書の詳細については佐藤ら (1987) を参照していただくとして　ここでの要点は　それが　(1) 共同利用メインフレームの上にあること　(2) 漢字コード体系が JEF コードと呼ばれる FACOM 独自のものであること　(3) 登録単語の総数が　地学用語約1万3千　地名関連約1万5千　計2万8千という膨大なものであることです．

というわけで　オリジナル辞書をパソコンに移植するには　いくつかクリアすべき関門があります．　どのような関門をどうクリアしたか順に述べてみましょう．

まず　最初の関門は　メインフレームからパソコンへのファイル転送です．

オリジナル辞書の大きさは　約5百キロバイトあります．　それだけの大きさのファイルをメインフレームからパソコン端末に誤り無しに転送するのは　簡単ではありません．　通常のターミナルエミュレーション等に用いられる公衆電話回線経由の無手順垂れ流し通信ではほとんど不可能です．　しかし　さいわいなことに　最近　物理探査部の中塚によって　ファイル転送プログラムが開発され (中塚　1987)　それを利用して　まったく誤りなしのファイル転送ができました．

ファイルがパソコンに移ったところで　次は　漢字コード体系の違いが重要な問題になります．

FACOM メインフレームは　JEF コードという呼び名の独自の漢字コードを採用しています．　一方　パソコンの MS-DOS は　シフト JIS コードと呼ばれる漢字コードを採用しています．　また　近ごろ　さかんに導入されている　unix ワークステーションでは　拡張 unix 漢字コード (16進表現は JEF 漢字コードと同じです．

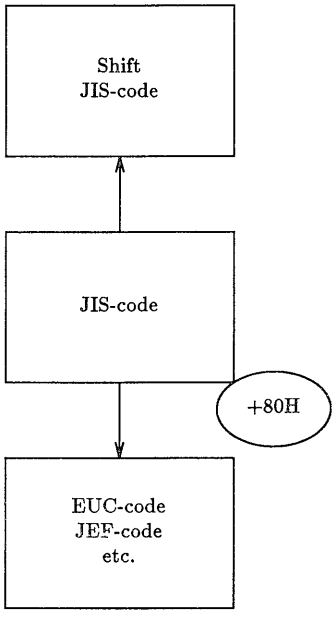

図4　各種漢字コードの間の関係．　シフト JIS 漢字コード　JEF 漢字コード　拡張 unix 漢字コード（EUC コード）等が JIS 漢字コードから派生していることをしめす．
JIS 漢字コード（2バイト）の各1バイトに80H（16進）を加えると JEF 漢字コードその他になる．
JIS 漢字コードから　シフト JIS 漢字コードへの変換は　少々こみいっているので　文献を参考にしていただきたい．

DEC 漢字コード　ATT 漢字コード等も同様です）が　多く採用されています（石田　1986）．　いずれにしろ　これらのいろいろな漢字コードは　JIS 漢字コードをもとにして　それを一定の手続きにしたがって変換して得ることができます．　したがって“かな”漢字対応表を広い範囲の計算機に移植するには　JEF コードから　JIS コードあるいはシフト JIS コードへの変換が必要です（漢字コードの変換については　図4　および　付録をご覧ください）．　JIS コードとシフト JIS コードとの変換については　井上ほか（1987）を　ご覧ください．

JEF コードから JIS コードへの変換は　各バイトから80H（16進の80　10進では128です）を引くだけの単純な作業です（図4）．　この　作業は　パソコンでおこないました．　これで　JIS コードによる“かな”漢字対応表ができました．

JEF コードからシフト JIS コードへの変換は　かなり複雑ですので　手続きの説明は省いて　付録に変換プログラムを載せておきます．　図5は　“かな”漢字対応表の一部を　16進数の JEF コード　JIS コード　シフト JIS コードで表現したものです．

このようにして　オリジナル辞書がパソコンの MS-DOS で利用できるかたちになりました．　最後に　辞書の大きさの問題を解決しなければなりません．

1．でもふれましたが　オリジナル辞書は　地名関係15,003単語　地学用語12,707単語　合計27,710単語を含む膨大なものです．　一方　代表的なワープロソフトである「一太郎」では　辞書の一括登録は7千語までに制限されています．　単語の削除が必要です．

オリジナル辞書では　変換効率を重視して　かなり長い複合単語が含まれています．　最長の単語は　「八方尾根超塩基性岩体」で　漢字10文字になります．　これらの　単語は　削除が可能です．　また「一太郎」付属の辞書を使って簡単に“かな”漢字変換できる単語も削除します（“簡単に”というのがクセモノですが　変換キーを2　3回押せば変換できるのを“簡単”ということにしています）．

この方針で　単語の削除を行った結果　地学用語は5千7百程度まで減らすことができました．　しかし残念ながら　地名関連用語は削除の余地が少なく　最終的に約1万2千単語となりました．　そのため　「一太郎 Ver. 3」用の“かな”漢字対応表としては　1で述べたとおり　(1) 地学用語（chigaku.dic）　5,683単語　(2) 地名（chimei. dic）　4,141単語　(3) 国土地理院発行

	べ	っ	と	ぶ	,	別	飛	,
ＪＩＳコード：	2459	2443	2448	2456	2124	4A4C	4874	2124
ＪＥＦコード：	A4D9	A4C3	A4C8	A4D6	A1A4	CACC	C8F4	A1A4
シフトＪＩＳ：コード	82D7	82C1	82C6	82D4	8143	95CA	94F2	8143

図5　パソコン用地学かな漢字対応表の一行を　JEF 漢字コード　JIS 漢字コード　シフトJIS 漢字コードのそれぞれで　どう表現しているか示す．　表記は16進数である．

図葉名（zumei. dic）　1,858単語　(4) 山川等の自然地名（yamakawa. dic）　5,689単語　の4つを作成しました.

4. 地学かな漢字対応表の提供について

以上　紹介してきました地学かな漢字対応表を　広く公開して　多くの地学関係者に使っていただきたいと思います.　提供できるのは　(1)メインフレーム用オリジナル辞書を図1の形式に変換して得られたファイル　(2)「一太郎」で　そのまま一括辞書登録できるように単語の削除を行ったファイル　です.　ファイルの形式は MS-DOS のテクストファイルで　1メガバイトフロッピーディスクが2枚になります.　MS-DOS システム　ワープロソフト　ワープロソフト用辞書等は　当然ながら含みません.

提供の方法　提供に際してのルール等については　所内で検討中ですので　詳しく述べられませんが　この地学かな漢字対応表の入手を希望される方は　担当者までご連絡くだされば　お答えします.

連絡先は　下記のとおりです.

郵便番号305　茨城県つくば市東1－1－3
工業技術院地質調査所地質情報解析室
野呂春文　村田泰章または佐藤岱生

引　用　文　献

B. W. カーニハン　D. M. リッチー（1978）プログラミング言語C（石田晴久　訳）　共立出版.

井上尚司　大野浩之　柳楽直樹　民田雅人　池田けんしろう（1987）: UNIX ワークステーション　NEWS, アスキー.

石田晴久（1986）: 日本語機能も定まった最近の UNIX 事情 bit, vol. 18, no. 6, 4-9.

萩谷昌己（1987）: GMW ウィンドウシステムについて　bit, vol. 19, no.3, 4-19.

中塚　正（1987）: ファイル転送ソフトウエア XTERM, RIPS（工業技術院共同利用研究情報処理システム）共用ソフト登録中.

桜川貴志（1987）: 開かれた日本語入力システム Wnn, bit, vol.19, no. 9, 13-23.

佐藤岱生・村田泰章・青木光子（1987）: 地学用かな漢字変換辞書の作成　地質ニュース　No. 394, 42-49.

付　録

```
/* jeftosj.c --  ＪＥＦ漢字コードのデータファイルを      *
*               シフトＪＩＳ漢字コードのファイルに      *
*               変換するためのプログラム                *
*               注意　２バイト漢字コード以外の処理は    *
*                     考慮していない                    */

#include        <stdio.h>

main()
{
        int     k, sk, d, c, cjis;
        long    count=0L;
        FILE    *from, *fto, *fopen();
        char    buffer[20];

                                                /* ファイルの準備 */
        printf(" ＪＥＦ漢字コードのファイル名 : ");
        scanf("%s", buffer);
        if((from=fopen(buffer, "rb")) ==0)      /* バイナリー */
          exit(printf("file %s not found¥n", buffer));   /* オープン */

        printf(" 出力ファイル名 : ");
        scanf("%s", buffer);
        if((fto=fopen(buffer, "wb")) == 0)      /* 同上       */
        {
          fclose(from);
          exit(printf("file %s can't open¥n", buffer));

        while((c = getc(from)) != EOF)          /* ファイルの終わりまで    */
        {                                       /* １バイトずつ処理する    */
                count++;
```

```
        if( c == 0x40 ) ;                       /* ＪＥＦの埋め草 を */
                                                /* 捨てる           */

        else if( c == 0x0a || c == 0x0d )       /* CR/LF はそのまま */
                putc(c, fto);                   /* ファイルに出力   */

        else
        {
                cjis = c - 0x80;                /* ＪＥＦ＝＞ＪＩＳ の変換 */

                if( count % 2 == 1 )            /* 第一バイトの処理 */
                {
                        k = cjis - 0x20;        /* 区数を求める */

                        if( k <= 62 )           /* 一区から六十二区なら */
                        {
                         if( k % 2 == 1 )       /* 奇数区なら */
                           sk = (k+1)/2 + 0x80;
                         else
                           sk = k/2 + 0x80;     /* 偶数区なら */
                        }
                        else                    /* 六十三区以上なら */
                        {
                         if( k % 2 == 1 )       /* 奇数区     */
                           sk = 0xe0 + (k-62)/2;
                         else                   /* 偶数区     */
                           sk = 0xe0 + (k-63)/2;
                        }
                        putc(sk, fto);          /* 第一バイトをファイルに */
                }                               /* 出力する             */

                else                            /* 第二バイトの処理 */
                {

                                if( k % 2 == 1 )        /* 第一バイトが */
                                {                       /* 奇数区なら   */
                                 st = cjis + 0x1f;
                                 if( st >= 0x80 )       /* 未使用領域の */
                                   st = st + 1;         /* 補正         */
                                }
                                else                    /* 第一バイトが */
                                 st = cjis + 0x7e;      /* 偶数区なら   */

                                putc(st, fto);  /* 第二バイトをファイルに */
                         }                      /* 出力する             */
                 }
        }

        fclose(from);                           /* ファイルを閉じて*/
        fclose(fto);                            /* 後始末          */
}
```

JEF 漢字コードをシフト JIS 漢字コードに変換するためのフィルタープログラム

JEF 漢字コードで表現されたテクストファイルを シフト JIS 漢字コードのテクストファイルに変換するための フィルタープログラムです.

このプログラムは JEF 漢字コードの処理のみ行います. 通常の EBCIC コードは 正常に処理されません.

標準的なC（カーニハン リッチー 1978）で記述してありますので 現在 市場に出回っているMS-DOS 用の C コンパイラーの大部分で コンパイルできると思います. unix マシンの C でも正常に動きます.