日本語ＵＮＩＸでの日本語入力方式と新機能
― ローマ字変換と漢字ヒストリ ―

中原 康（東芝・青梅工場）

１．はじめに

東芝が日本初の日本語ワードプロセッサを世に出してから５年、計算機の日本語処理は非常に身近なものとなった。日本語処理が普及するにつれ、日本語ワードプロセッサの文書作成専用エディタ以外でも、様々なプログラムで日本語処理が必要になって来ている。

日本でのＯＡやワークステーションなどを考えるとき、どうしても日本語の取り扱いが大きな問題であり、ベースとなるＯＳに日本語処理機能をどう組み込むか悩むところである。中でも日本語入力の問題が最も大きなものであろう。

ところで、世界的な標準ＯＳとして定評のあるＵＮＩＸには、従来のＯＳにない優れたマン・マシーンインタフェースを提供するシェルやユニークで有用な多くのコマンド群があり、ソフトウェア開発やドキュメント作成環境などＵＮＩＸに精通すればする程、その素晴らしさには目を見張らせられる。

東芝では、ＵＸ－３００（OS/UX V1）において、こうしたＵＮＩＸの設計思想を損うことなく、自然な形でＵＮＩＸにＯＳレベルで、文法処理機能を持ったかな漢字変換を含め、日本語処理機能を追加した。更にその親和性の高い日本語処理方式の利点を生かして、ＵＸ－３００Ｆ（OS/UX V2）では、日本語入力機能の強化拡充と、シェルなど上位ソフトの日本語処理を充実させ、“日本語ＵＮＩＸ”としての形態を整えるまでになった。本稿では、その日本語入力方式と、操作性向上のために追加した独自のローマ字かな漢字変換，ユーザ定義可能な拡張ローマ字変換や漢字キー機能，高頻度使用漢字の高速入力を行う漢字ヒストリ機能について報告する。

２．ＵＸ－３００の日本語入力方式

一般に、計算機に日本語入力を組み込む場合、

① 専用システム方式，
② サブ システム方式，
③ サブルーチン（ライブラリ）方式，
④ ＯＳ標準入出力方式

が考えられる。①②は、ワードプロセッサに多く見られる方式で、③④は、一般プログラムでの日本語処理向きの方式である。

ＵＮＩＸのような環境に馴染ませるには、①②は、システム切り替えやファイル変換が必要となり、更に専用の日本語エディタを経由しない日本語入力が不可能となる点で、また③は、既存のプログラムでの日本語入出力が不可能であり、またライブラリを意識的に使い分けたプログラムでしか日本語入出力が可能とならない点などで不向きである。

④のＯＳ標準入出力方式といっても様々なレベルがあり、端末自身に漢字変換機能を持つものや、一部のマイコン，ミニコン、オフコンでは、ＯＳ自身に漢字変換機能を既に組み込んでいるものもある。しかし、利用者の側から見れば、かな漢字変換と言っても音訓表程度のものであったり、日本語を自然に入力するには、ワードプロセッサに比べて著しく能力が低かったり、またサブシステムとし

ＵＮＩＸは、米国AT&T BELL研究所の登録商標である。

て備わっているワードプロセッサ自身、文法処理機能を持たないものもあり、まだまだ使い勝手の良いものは少ない。

一方、ワードプロセッサでの日本語入力は、べた書き・複文節処理のかな漢字変換へと重点が移って来ている。しかし、この変換方式には、文節の認識問題を含め、変換率など残された問題も多く、専用エディタ以外での日本語入力には向いていない面が多い。

以上のような観点から、ＵＸ－３００では、④のＯＳ標準入出力方式を採用し、かな漢字変換としては、文法処理機能を持った漢字指定方式のかな漢字変換を採用した。漢字指定方式としたのは、入力を受けるプログラムが日本語専用エディタのような同音異義語の選択や文節識別能力を持つことを前提とできないことや、べた書き下での余計な文法処理が、かえって自然な入力の妨げになり得ることを考慮したためである。次に述べるように、同音異義語の選択操作もＯＳ側で行っている。ユーザプログラムに余計な選択操作を強いないようにするためである。

ところでＵＮＩＸの入出力の特徴には、次の３つがあり、

1) 全二重 I/O,
2) data-driven I/O,
3) stream I/O

一方日本語、特に漢字の入力の特徴として、入力過程に、

ｱ) 辞書の参照,
ｲ) 同音異義語などの選択操作

が介在することが挙げられる。このため、ＵＸ－３００ではＯＳ核と一体になって動作する漢字デーモンプロセスが存在する。一般の非漢字端末からの入力は、そのままユーザプログラムに渡されるが、漢字端末からの入力は、一度この漢字デーモンプロセスで受け、漢字への変換とその同音異義語などの選択処理が終ったものがユーザプログラムに渡される。（ＵＮＩＸ流に言うと、この漢字デーモンはキーボード入力の画面表示，かな漢字変換，候補漢字の選択を行う“フィルタ”群から成り、各々独立したモジュール構造を持っていて、プログラム部品として独立している。端末特性に依存するのは、表示部と選択操作部に局所化されている。）

このように、ＵＸ－３００の日本語入力方式は、全体として、ＯＳ標準入力での“逐次変換・逐次選択方式”の“文法処理機能付き漢字指定方式”のかな漢字変換となっている。

3. ＵＸ－３００のかな漢字変換

ＯＳ標準に組み込んだかな漢字変換と言っても、ワードプロセッサに比べて機能が劣っていたのでは、システムとして一貫した高度な日本語入力インタフェースを実現したことにはならない。やはり、漢字変換能力とその操作性が重要であろう。文章やデータとしての幅広い日本語の入力を統一したインタフェースで実現することが大切である。このため、ＵＸ－３００では、次のような漢字変換方式を採用した。

3.1 かな漢字自動変換

文章を自然な形で入力するには、やはり文法解析能力が必要である。完全な複文節のべた書き入力が理想ではあるが、前述のように現状では、変換すべき漢字部分を入力者が指定する“漢字指定のかな漢字自動変換”が最も適している。

（ここで「自動」というのは、文法処理能力を有していて、活用語の入力も終止形などではなく、活用形そのままの読みで入力できるという意味で用いている。）
ＵＸ－３００では、単一名詞の他に、活用語・送りがな・助詞・助動詞の認識、接頭・接尾語や助数詞・漢数字の変換機能を有している。

入力は、漢字部の前に「かな漢字」キーを押し、読みをひらがなで入れ、送りがなどのひらがなは、その前に「ひらがな」キーを押してから入力する。画面上では、漢字部が【 】で囲まれる。

（例）

・【うつく】しい【にほん】の【しぜん】
　→　美しい日本の自然
・【だい１０かい　たいかい】で【こう・せいせき】を【あ】げた
　→　第十回大会で好成績を上げた

3.2 固有名詞変換

日本語入力では、地名・人名などの入力も重要である。このため、かな漢字自動変換で用いる一般辞書（国語辞典に相当）の他に、固有名詞辞書を設け、地名・人名・会社名などをその読みから漢字に変換する固有名詞変換がある。入力は、「かな漢字」キーの代りに「固有名詞」キーを使用することを除いて、かな漢字自動変換と同じである。 表示は、漢字部が〖 〗で囲まれる。

（例）

・〖とうきょうと　みなとく〗の〖かわさきいちろう〗さん
　→　東京都港区の河崎一郎さん

3.3 一字変換

かな漢字自動変換や固有名詞変換で変換できないような漢字１字１字について、その音読みや訓読みから目的の漢字に変換するのが一字変換である。音訓表による入力である。読みとして２つまで指定できる。また、訓読みの送りがなは「－」で区切って指定することができる。入力は、「一字変換」キーを押し、読みをひらがなで入力する。 表示は、漢字部が◆で囲まれる。

（例）

・◆いち　じ　へん・かーわる　かん◆
　→　一字変換

3.4 区点変換

上記のひらがなの読みからの変換では変換できない特殊記号などを、ＪＩＳコードの区点番号や16進コードを直接指定することで目的の漢字に変換する。このコード入力は一般に、入力するコードを知っている必要があるが、コード全桁を知らなくても一部の桁（例えば区番号）だけを指定するとシステムで候補漢字の表示を行うようになっていて、“選択入力”も可能である。入力は、「区点変換」キーを押し、コード（数字，英字）を入力する。16進コードは「ｘ」を先行させる。不明な桁は、「？」で指定したり、不明な桁で終る場合は省略することもできる。表示は、対応部が◇で囲まれる。

（例）

・◇０６３３　ｘ２６４２　０６？？◇はギリシャ【もじ】です
　→　αβγはギリシャ文字です

3.5 候補漢字の選択

前述の漢字変換は、一般に「ＥＸＥＣ」キーの押下によって変換が実行され、その結果同音異義語などの選択が必要なものは、逐次その選択を行うようになっている。これは、日本語入力がＯＳ標準入力として行われるため、ワードプロセッサの専用エディタのような選択一括処理がユーザプログラムとの間では、実現できないからである。（必要とあれば、ユーザプログラムにこの候補漢字すべて渡すこともできるが、専用エディタでもない限り、ユーザプログラムにこれを送り込んでも、あまり意味がない。事実、ＵＸ－３００上の日本語テキストエディタや日本語スクリーンエディタでも同音異義語の選択処理などはすべてこのＯＳ標準の選択処理に委ねており、エディタ本来の処理に専念している。）

ところで、同音異義語の選択操作は意外に厄介なものである。ＵＸ－３００では、辞書に設定された漢字の使用頻度の順に表示が行われるが、使用者によって違いのあるのが普通である。このため、更に使用者（端末）ごとに暫定辞書を設けて、かな漢字自動変換や固有名詞変換では、最新に選択された漢字を最優先に表示する機能も持っている。

3.6 画面表示

ＵＸ－３００で使用する漢字端末は、漢字コード(JISC6226)と英数コード(JISC6220)との混在が可能となっている。キーボードには、「漢字」モードキーや各種の漢字変換キーがあり、それらの押下によって、“漢字モード”と“英数モード”が切り替るようになっている。画面は、漢字で 40文字 × 40行、英数字で80文字 × 40行の表示が可能である。日本語入力は、この画面下方の３行を用いて行われる。漢字端末が“漢字モード”になると、カーソルは画面下方のシステム行に移り、ここで、かな漢字入力や候補漢字の表示・選択が行われる。選択の終った文字は順次画面上方に移る。（同時にユーザプログラムに渡される。）“英数モード”になると、システム行は解放され、カーソルは上方のユーザ行に移る。

4. ローマ字変換

前述の漢字変換などの日本語入力は、ＪＩＳひらがな配列に漢字変換・選択用の制御キーを加えたキーボードを用いて行うが、ひらがなキーや変換制御キーの使用は慣れないと大変である。ひらがなキーも含めてブラインドタッチができることが理想だが、プログラマなどのように日頃英数キーに慣れていて、なかなかひらがなキーの操作が苦手の人には、ローマ字入力が便利である。

ローマ字入力と言っても様々なものがあり、ワードプロセッサにおいても夫々独自な方式でサポートされているが、マン・マシーンインタフェースを考えると、入力し易さは勿論のこと、入力した文字の確認のし易さや変換・選択操作への配慮も重要である。ＵＸ－３００Ｆでは、こうした観点から前述のすべての漢字変換に対して、ローマ字→かな漢字変換ができるよう次のような仕様のローマ字変換を付加した。

4.1 つづり方

ローマ字のつづりは、訓令式を基準に、日本式やヘボン式も可能とし、加えて「っ」，「ょ」などの単独よう音や「ディ」，「ティ」などの外来語用のカタカナなどを入力し易くするために若干のローマ字つづりを追加した。

(1) ひらがな/カタカナの切り分け

小文字/大文字で始まるローマ字を、各々ひらがな/カタカナに変換する。これは、固有名詞などを大文字で書くとする一般のローマ字つづりと異なるが、以下に述べるように、入力はローマ字，表示は ひらがな/カタカナを基本的な考え方とし、ローマ字を入力手段と考えるためであって、できるだけ“モード”切り替えキーを設けないことを前提とした。

(2) はつ音「ん」の取り扱い

「ｎ」を用いるが、「ｂ・ｍ・ｐ」の前では、「ｍ」も使用できる(ヘボン式)ようにした。

(例) ronbun = rombun = ろんぶん

また、「ん」を表わす「ｎ」に「ａ・ｉ・ｕ・ｅ・ｏ および ｙ」が続く場合には、音節の区切りを示すために「'」を用いる。

(例) tan'i = たんい

(3) 促音「っ」の取り扱い

次に続く子音を重ねるが、「ｃｈ」の前では「ｔ」も使用できる(ヘボン式)。

(例) itti = itchi = いっち

(4) よう音の取り扱い

正しいつづりであれば子音により自動的に生成されるが、単独で指定する時は、「ｘ」または「￥」(ＵＮＩＸの一般的なエスケープキャラクタ)を先行させる。

(例) xa = ¥a = ぁ　xi = ¥i = ぃ　xu = ¥u = ぅ
xe = ¥e = ぇ　xo = ¥o = ぉ
xtu = ¥tu = っ　xtsu= ¥tsu= っ
xya = ¥ya = ゃ　xyu = ¥yu = ゅ　xyo = ¥yo = ょ
XKA = ¥KA = ヵ　XKE = ¥KE = ヶ

(5) 長音の取扱い

長音「ー」はそのまま「ー」として入力されるが、母音の後に同じ母音や「^」，「¯」を続けると次のように変換される。

aa = ああ	a^ = あ^	a¯ = あ¯	a- = あー
ii = いい	i^ = いい	i¯ = いい	i- = いー
uu = うう	u^ = うう	u¯ = うう	u- = うー
ee = ええ	e^ = えい	e¯ = えい	e- = えー
oo = おお	o^ = おう	o¯ = おう	o- = おー

(例) to^kyo^ = to¯kyo¯ = toukyou = とうきょう
yu^e^ = yu¯e¯ = yuuei = ゆうえい

4.2 ローマ字変換モードの設定

ローマ字入力は、端末が“漢字モード”の時に有効で、その中で利用者が任意にローマ字変換モードをオン・オフできるようになっている。ローマ字変換モードにするには、「ＰＦ９」のファンクションキーを使用する。なお、このキーは

ローマ字変換モードをオン・オフするトグルキーになっている。

4.3 ローマ字入力の表示

入力はローマ字で行うにしても、やはりその表示はひらがなやカタカナで行われた方が便利である。入力ミスも早く発見し修正することができる。そのため、入力されたローマ字は、音節単位で直ちに、ひらがなやカタカナで表示する方式を採用した。音節確定までの途中の子音は最高４文字まで、特定エリアに蓄積・表示され、子音の入力ミスにも直ちに対応できるようになっている。

4.4 ローマ字入力中での英字入力

ローマ字入力中に、英字を入力するには、ローマ字変換モードをオフする(4.2)か、「ＣＴＲＬ＋Ａ」キーの押下により一時的に“英字シフト”にしてから行う。

この「ＣＴＲＬ＋Ａ」キーもトグルキーである。

4.5 記号などのローマ字入力

日本語入力を行う上で、句読点や様々な記号が必要となることが多い。そのために次のような記号変換と拡張ローマ字変換を採用した。

, （コンマ）	、（読点）
. （ドット）	。（句点）
－ （マイナス）	ー（長音）
lh (left hook)	「（左鍵括弧）
rh (right hook)	」（右鍵括弧）
ldh(left double hook)	『（左二重鍵括弧）
rdh(right double hook)	』（右二重鍵括弧）
lq (left quote)	‘（左引用符）
rq (right quote)	’（右引用符）
ldq(left double quote)	“（左二重引用符）
rdq(right double quote)	”（右二重引用符）
cm (comma)	，（コンマ）
pd (period)	．（ピリオド）
dt (dot)	.（ドット＝ピリオド）
md (middle dot)	・（中点）
br (bar)	―（バー）
wv (wave)	～（波形）
ws (white star)	☆（白星印）
bs (black star)	★（黒星印）
wc (white circle)	○（白丸）
bc (black circle)	●（黒丸）
dc (double circle)	◎（二重丸）

なお、「，」や「．」，「－」などをそのまま本来の文字として入力するには、ＵＮＩＸ流に、「￥」を先行させる。

拡張ローマ字変換というのは、ローマ字のつづりとしては使用されない４文字

までの子音の組み合せに、１文字の漢字コードを対応させるもので、大文字でも小文字でもよい。上記はシステムとして標準でサポートされるものである。しかし、この種のものは何の標準もなく、利用者の好みも多いところでもあり、また対応させる漢字を、第１水準や第２水準の漢字とすることなど、幅広い使い方が考えられる。

ＵＸ－３００Ｆでは、このため拡張ローマ字変換を利用者が必要に応じて端末ごとに任意に設定できるようになっている。それには、まず

子音ニーモニック　　変換される漢字１文字　［コメント］

の行からなる拡張ローマ字変換定義ファイルを作成する。ニーモニックとしては、大文字の子音２～４文字を用いるが、先頭に「Ｎ」は使えない。また、標準ローマ字つづりの接頭辞や定義ファイルの中で互いの接頭辞になるものも禁止される。ニーモニックと漢字文字との間は空白かタブで区切る。漢字は、漢字変換用の制御記号の「【】〖〗◆　◇」以外なら何でもよい。最大１２８個定義可能である。

ファイルができると、コマンド“uroma”を用いてシステムに切り替え要求を出す。

```
$ uroma my_romaji        ( my_romajiが定義ファイル名である。)
```

この“uroma”コマンドでは、設定する端末を指定することもできるが、これは、スーパーユーザ（システム管理者）に限られる。普通は、使用している端末の拡張ローマ字変換が切り替わる。また“-p”オプションで現在その端末に設定されている拡張ローマ字変換の一覧を見ることができる。更に、このコマンドは以前に使われていた拡張ローマ字変換のファイル名を返すので、シェル変数として取り込み、後で前の拡張ローマ変換ファイルに戻すこともできる。

4.6 ローマ字入力での漢字変換制御キー

ローマ字入力の考え方には、できるだけ英数配列のキーで、日本語入力が行えるようにとの思想がある。ひらがなやカタカナの入力が楽になったからといって、肝腎の漢字変換操作が楽にならなければ余り意味がない。ＵＸ－３００では、前述したように、漢字部指定方式のかな漢字変換を採用しているので、これに合せて漢字部を指定する制御キーを、次のように英記号キーで代替えできるようにした。これら英記号をそのままの文字として入力するには、「¥」を先行させる。

［（左ブラケット）	【（かな漢始め）
］（右ブラケット）	】（かな漢終り）
｛（左ブレース）	〖（固有名詞始め）
｝（右ブレース）	〗（固有名詞終り）
”（ダブルクォート）	◆（一字変換始め／終り）
％（パーセント）	◇（区点変換始め／終り）

また、変換の実行や変換後の選択処理用の制御キーも、次のようなキーで代替えできるようにしている。

RETURN	EXEC キー（ただし、先頭文字の時を除く）
	選択モードでは、次候補（いいえ）キー

BS	選択モードでは、前候補(←)キー
スペース	選択モードでは、選択(はい)キー

4.7 ローマ字入力の例

ローマ字入力を使ったかな漢字変換の例を挙げる。

・ [RO-MAmdji nyu^ryo^ku]ha,hiraganaEKO-sareru.

→ 【ローマ・じ にゅうりょく】は、ひらがなエコーされる。

→ ローマ字入力は、ひらがなエコーされる。

5. 漢字ヒストリ

文章なりデータなり、日本語入力を行う場合、同じ漢字を何度も入力することがある。現に、今回日本語ＵＮＩＸのマニュアル：日本語入出力編など、この日本語システムを用いて作成したが、当初、“さっき”の漢字がここでもう一度使えたらなど、漢字の再入力に良い方法はないものかと痛感させられた。

漢字の入力は、(ローマ字→)かな→漢字変換→選択操作が必要なので、非常に厄介である。同音異義語の表示に頻度処理や学習機能があり、一方“新語機能”によって短い読みで高頻度の漢字を登録して置くこともできるが、いずれにせよ毎回、漢字変換と選択操作が必要で、効率が悪い。

このため、利用者ごとに漢字変換・変換された漢字を記憶して置き、その再入力は、ファンクションキーを用いて一気に漢字そのものを入力する“漢字ヒストリ”機能を付加した。これは、ローマ字/ひらがな入力や漢字変換，選択操作の過程を経ないので、非常に効率的な入力ができる。

5.1 漢字ヒストリの記憶単位

漢字ヒストリに記憶されるのは、すべての漢字変換(一字変換や区点変換も含む)で選択された文字であり、漢字コードであれば記号などであっても、一部の漢字変換制御記号「【】〖〗◆◇」を除いて何でもかまわない。更に、記憶の単位は、変換フィールド全体、つまり複合語単位である。例えば、

【たんご】【たんい】 → 単語単位

と変換した時は、「単語」と「単位」が別々の漢字ヒストリとして記憶されるが、

【ふくごう・ご たんい】 → 複合語単位

と変換した時は、「複合語単位」が１つの漢字ヒストリとして記憶される。

5.2 漢字ヒストリの個数と制御

漢字ヒストリは何個ぐらいが適当だろうか？当初は、利用者(端末)ごとに１０個としていた、次に述べるように表示の問題もあったからである。しかし、小説や新聞の社説、紀行文、随筆、自然科学の解説文などを調べ、およそ３０個が適当と考えられ、この数値を採用している。その制御については、画面表示の都合から、最新の１０個を“フロント”の漢字ヒストリとし、後述するようにこれは、漢字ヒストリの“モニタ表示”の対象となる。このフロントの漢字ヒストリは、ＬＲＵ(Least Recently used)で管理され、そのＬＲＵが“バック”の２０個の漢字ヒストリに回される。このバックの漢字ヒストリについてもＬＲＵ制御が行われる。

5.3 漢字ヒストリの表示と入力

漢字ヒストリは、通常表示されない。次の操作で見ることができる。

1) フロントの１０個を表示　　「ＰＦ１」+「空白」+「空白」
2) 全ヒストリを表示　　「ＰＦ１」+「空白」+「！」
3) 指定のヒストリのみを表示　　「ＰＦ１」+「空白」+「番号」

1)は、システム第３行に１行で、番号付きで表示されるが、３文字以上のものは、２文字とその後に、「*」が補われる。2)は、ユーザ行に全ヒストリの全文字が、表示される。3)は、システム行で指定のヒストリを全文字表示する。

一方漢字ヒストリからの入力は、次のように“ヒストリ番号”を用いて行う。

a) フロント(0～9)の中から　　「ＰＦ１」+「１桁番号」
b) 全ヒストリ(01～30)の中から　　「ＰＦ１」+「！」+「２桁番号」

漢字ヒストリを利用して入力した漢字は、直ちにシステム第２行(漢字入力行)に“漢字エコー”される。間違った場合は、この段階で普通のひらがな/カタカナと同じように修正キーで修正もできる。変換実行により、この漢字はそのままユーザ行に移り、プログラムの入力となる。

このように、漢字ヒストリの入力には、そのヒストリ番号を利用者が知っておく必要がある。毎回表示要求を行って確認していたのでは効率が悪い。そのため、フロントの漢字ヒストリを常時システム第３行に表示して置く“漢字ヒストリのモニタ表示”機能を設けている。指定は、“漢字モード”で「ＣＴＲＬ+Ｐ」のキーで行う。なお、これもトグルキーとなっている。

漢字ヒストリのモニタ表示モードでは、漢字ヒストリに新たな漢字が追加されるごとに表示が行われ、選択した漢字が何番の漢字ヒストリに登録されたか直ちに確認が取れるので便利である。

5.4 漢字変換としての漢字ヒストリ

バックの漢字ヒストリを入力するには、前述のようにバックのヒストリ番号を知らなくてはならない。表示要求による確認もできるが、３０個の同音異義語をもつ漢字の選択とも考えられる。漢字変換・選択操作を省くための漢字ヒストリ機能ではあるが、苦労して長い複合語などを選択した後で、自分の最近使った中から選択できるだけでも、効率は良いものである。このような観点から、漢字ヒストリの中から、目的の漢字を選択する次のような“漢字ヒストリ変換”も可能とした。指定は、「ＰＦ１」の後に変換制御キーを押す。

「ＰＦ１」+「ＥＸＥＣ」　(ローマ字モードでは、「RETURN」も可)
「ＰＦ１」+「次候補」　(ローマ字モードでは、「RETURN」も可)
「ＰＦ１」+「←」　(「BS」も可)

この操作で、一字変換と同じようにシステム第３行に１０個ずつ、漢字ヒストリが表示され、後は、一字変換と同じ候補選択操作で目的の漢字を選ぶ。

6. ユーザ漢字キー

使用頻度の高い漢字(文字列)や定型句，会社名を効率良く入力するには、前述の漢字ヒストリ機能や新語機能(辞書登録)があるが、更に効率良く特定の漢字列を入力できるよう、ＵＸ－３００Ｆでは、ユーザ定義可能な漢字キー機能を設けた。これは、利用者(端末)ごとにユーザの設定した漢字列を特定のキーに任意に

割り付けるものである。割り付けは、定義順に番号を振り、その入力は、ファンクションキー「ＰＦ２」＋「番号」で行う。

6.1 ユーザ漢字キーの定義

ユーザ漢字キーを定義するには、まず、

変換する漢字列　［コメント］

のような行からなるユーザ漢字キー定義ファイルを作成する。この各行先頭の漢字列が、順次その番号の“ユーザ漢字キー”に対応する。最大１２８個である。漢字列としては、漢字変換制御用の「【】〖〗◆◇」を除いて漢字コードなら何でもよい。ファイルの作成が終ると、コマンド“ukk”を用いてシステムに登録（変更）要求を出す。

```
$ ukk my_kanji          (“my_kanji”が定義ファイル名である。)
```

この“ukk”コマンドでは、設定する端末を指定することもできるが、これは、スーパーユーザ（システム管理者）に限られる。普通は、使用している端末のユーザ漢字キーが切り替わる。また“-p”オプションで現在その端末に設定されているユーザ漢字キーの一覧を見ることができる。更に、このコマンドは以前に設定されていたユーザ漢字キー定義ファイル名を返すので、シェル変数として取り込み、後で前のユーザ漢字キーに戻すこともできる。

6.2 ユーザ漢字キーの表示と入力

前述の漢字ヒストリキーの場合と、対応する漢字が動的に変化する/しないの違いを除いて、両者の差はない。したがってその表示や入力の仕方も、漢字ヒストリの「ＰＦ１」を「ＰＦ２」に置き換えるだけで、あとは全く同じである。

7. おわりに

以上、日本語ＵＮＩＸ：ＵＸ－３００Ｆの日本語入力について、その方式や新機能から報告した。日本語入力の方式や操作性に様々な工夫を行い、ワードプロセッサ以外でのＯＳ標準入力から漢字変換を考えているのが大きな特徴である。

日本語処理の普及を考えると、まだまだ日本語入力に残された問題は多いと思われる。この方式や新機能の評価も含め、今後共、改善・改良を加えて行きたい。

最後に、日本語ＵＮＩＸ開発の機会を与えられ、日頃からご指導いただいているソフトウェア第３部の開発部長、三好課長、また今回の開発にあたり討論や協力をいただいた関係者各位に感謝の意を表する。

【参考文献】

[1] 三好，中原 ： 「UNIX の日本語処理」，bit Vol.15 No.5，1983
[2] 中原 ： 「UX-300 での日本語処理」，情報処理学会第27回全国大会，1983
[3] 東芝マニュアル ： 「TOSBAC UX-300 日本語入出力機能説明書」，東芝，1984

（なお、本稿はＵＸ－３００Ｆで作成したものである。）