

Web パブリッシングにおけるアジア諸言語の文字情報

アジア諸言語語学教育 CAI のための文字表示システム開発を通じて

永崎 研宣*

1 はじめに

WWW(World Wide Web) においてアジア諸言語の多様な文字による情報を共有する方法は様々に模索され、いくつもの方法が提案されている。しかしながら、それらはどれも一長一短であり、現状の様々な環境の大半において容易に利用可能な方法が方法が確立されているとは言い難い状況である。アジア諸言語のための語学教育 CAI プログラムの作成に携わっていた筆者が必要としたのは、データの編集が容易で、短い単語や文章の表示にあたってはあまり問題にならないような表示/実行速度を備えた方法であり、かつ、多少古い環境でもうまく動作するような方法であった。これにうまく適合するものが提供されていなかったため、こうしたニーズをうまく満たした、アジア諸言語の文字による情報を共有するためのシステムを独自に構築したのである。本稿では、現状において利用可能な、WWW におけるアジア諸言語の文字情報共有の手法について概観した上で、筆者が構築した多言語文字情報共有システムを紹介し、それを通じて、WWW におけるアジア諸言語の文字情報共有の現状と今後について検討してみたい。

1.1 コンピュータと多言語文字環境

近年、コンピュータ上で多言語文字情報を処理する仕方は、だいぶ容易になってきた。Microsoft Windows を用いればある程度までは簡単にできるようになっており、また、Apple 社では Language kit という形で¹⁾ それを実現している。また、早稲田国際化・多言語環境 (System 1)²⁾ や超漢字³⁾ など、様々な方面からの試みもなされてい

* 研究協力者: 東京外国語大学アジア・アフリカ言語文化研究所 COE 非常勤研究員

1) MacOS 9 に付属する Language kit でサポートしているのは、アラビア語、ヘブライ語、デーヴァナーガリー文字、中央ヨーロッパ諸語、キリル文字、グジャラティー文字、パンジャビー文字、韓国語、中国語 (繁体字・簡体字) となっている。さらに、これ以外にも、たとえば、チベット語では Tibetan Language Kit for MacOS がある。<http://www.toyo-bunko.or.jp/Tibetan/WS/TibetanLanguageKit.html>

2) System 1 は、早稲田大学が中心となって開発されてきたものであり、現在のところ、UNIX 等で主に用いられている X window 環境向けに実装がすすめられている。

<http://mling1.kake.info.waseda.ac.jp/system1/>

3) 超漢字 3 は、TRON プロジェクトの一貫として産み出された Operation System で、GT 書体フォント (GT 書体フォントは日本学術振興会の未来開拓学術研究推進事業「マルチメディア通信システムにおける多国語処理の研究」プロジェクト (<http://www.l.u-tokyo.ac.jp/GT/>) の成果である。なお、超漢字 2 では今昔文字鏡 (<http://www.mojikyo.org/>) を採用していた) などを取り込むことで多くの

る。また、アプリケーションとしても様々なものが開発されている。GNU プロジェクトが開発している GNU Emacs ⁴⁾ は、メールや WWW も含めた多言語環境を可能とするエディタとして有名である。UNITYTYPE ⁵⁾ の開発している Unitype Global Writer は、100 以上の言語に対応している多言語ワープロソフトである。さらに同社は、Microsoft Office の多言語対応アドオンソフトとして Unitype Global Office もリリースしている。

2 WWW におけるアジア諸言語文字情報の共有

ここまでみてきたように、コンピュータ上でアジア諸言語を含む多言語文字環境を実現する方法は、どれも完全とはいえないにせよ、すでに様々な方法が提供されている。しかしながら、WWW (World Wide Web) を利用した Web パブリッシングにおいてはやや事情が異なる。

2.1 前提となる環境について

WWW は Web ブラウザを介してインターネット上で情報を共有する仕組みであり、HTML を用いるということ意外には特に規定はなく、HTML を用いるということですから強制力を持っているわけではない。すなわち、様々な異なった環境が共存しているということを前提としなければならず、そこで情報を発信しようとするなら、それらの中から最大公約数的なものを見いだしていかなければならない。多言語文字情報を現存する Web ブラウザの中でも比較的によく使われているものには、Internet Explorer、Netscape Communicator、Netscape6、Mozilla、lynx、w3m、GNU Emacs などがある。⁶⁾

また、Web ブラウズに利用される OS としては、MS-Windows、Macintosh、Linux といったところが優勢である。⁷⁾

こうしたことから、大勢としては、MS-Windows 環境で Internet Explorer か Netscape の利用を想定しておけばよく、多少マイナーな環境に配慮するとしても、OS として、Linux、Macintosh まで考えておけばかなりの部分を網羅できることになる。したがって、本稿では、Web ブラウザとしては Internet Explorer 及び Netscape を、OS とし

文字を使える環境を実現している。(http://www.chokanji.com/) 超漢字 3 のアプリケーションとして「超漢字ウェブサーバ」というものが発売されているが、これについては後述する。しかし、「アラビア語やヘブライ語など、特殊な書記方向の言語には対応しておりません。文字単位でのご利用になります。とのことである。この点については、超漢字 2 への言及ではあるが、「超漢字管見」(http://www.horagai.com/www/den/kankenN.htm)などを参照。

⁴⁾ <http://www.gnu.org/software/emacs/emacs.html>

⁵⁾ <http://www.unitype.com/unitype.htm>

⁶⁾ たとえば、筆者が個人的に運営する人文科学系学術向けのサイト、philosophy.onweb.to における 2001 年 3 月のアクセスログを見るなら、36588 件中、MS-IE 21027 件 (57.5%) Netscape(Mozilla) 10209 件 (27.9%) JAVA の利用が可能な Web ブラウザで 85 その他の中には、サーチエンジンのロボットによるアクセスも多く含まれている。また、WebSideStory の一部門で Web アクセスデータの分析を行っている Statmarket の 2001 年 2 月 21 日の調査によると、IE が 87.71% (<http://statmarket.com/SM?c=stat022201>)

⁷⁾ 前出の筆者のサイトのアクセスログでは、36588 件中、MS-windows 23750 件 (64.9%)

ては、MS-Windows、Linux 及び Macintosh を主に念頭に置くことにする。

2.2 Web パブリッシングにおけるアジア諸言語文字情報の方法について

Web パブリッシングにおいてアジア諸言語の文字情報を共有する方法には、すでに様々なものが用意されている。情報受信者側で特別なものを用意しなくともアジア諸言語の文字情報を共有できる仕組みとして現在利用されている主なものとしては、

- a. Font のダウンロード b. Font Embedding(PDF を含む) c. JAVA Applet を利用する
- d. 文字を画像として表示する

といったものが挙げられよう。⁸⁾

しかしながら、筆者の携わっていたアジア諸言語向け語学教育 CAI プログラム⁹⁾ において求められていた、

- 問題データファイルをエディタやワープロで簡単に編集できる。
- WWW からでも問題データの作成ができる。
- 性能の低いコンピュータでも比較的にはよく動く¹⁰⁾
- なるべく様々な環境で動作できる

といった条件を現状において満たし得るのは d. のやり方だという結論に達したのだが、まずは以下に一つずつ検討してみよう。

a. Font をダウンロード

言語ごとに専用のフォントをダウンロードするという方法である。¹¹⁾ この方法は、同じ環境同士ならば完全に問題なく利用可能だが、OS 等が異なった場合、インストールできないといった問題が生ずることがある。また、そのフォントを用いてデータを編集しなければならないため、編集の際、そのフォントに対応したエディタやワープロソフト等を用意しなければならない。今回の CAI プログラムでは、問題データを通常のエディタやワープロソフト等でも容易に扱えることを目的としていたため、この方法は採用しなかった。

⁸⁾ 1997 年 8 月に出版された三上吉彦・関根謙司・小原信利『マルチリンガル Web ガイド』(オーム社)では、当時行われていた様々な多言語 Web ページ構築の方法が紹介されている。Font をダウンロードする方法のみならず、すでに、JAVA Applet を利用したものや PDF を利用したもの、PostScript を利用したものなど、様々な方法が具体的に挙げられている。

⁹⁾ この語学教育 CAI プログラムは、筑波大学(当時)の山元啓史氏が開発していた、Web 利用型のドリル形式のものであり、ddf(drill data format)と呼ばれるテキストベースの簡易なデータフォーマットを用いることで、問題作成をエディタやワープロ等でも容易にできるようにしたところの一つの特徴がある。筆者は東京外大 AA 研峰岸真琴氏の指導の元、このプログラムをアジア諸言語の文字に対応させ、かつ、コンピュータにさほど詳しくない人でも、問題作成を Web からでもワープロ・エディタ等でも容易に行えるように改良を加えたのである。これについての詳細は別稿を期したい。

¹⁰⁾ これは、具体的に数値を挙げることはできないが、件の CAI プログラムを、新しいコンピュータを導入する機会のないところでも満足に使えるようにしたいとの考えからである。

¹¹⁾ 筆者の知る限りでは、特にインドのサイトではこの方法を用いているところが未だに少なくない。インドでは、互いに互換性のない様々な文字コード/フォントが存在することがその一つの理由だと思われる。

b. Font Embedding

Web ページにフォントを埋め込むという方法には Microsoft の Web Embedding Fonts Tool¹²⁾ と Netscape の Dynamic Fonts¹³⁾ がある。これらの技術は、当初は、より自由度の高いデザインをしたいというニーズに基づいて、情報受信者側に存在しないフォントでも自由に利用できるようにしたものなのだが、まったく異なる言語のフォントを埋め込むことも可能であったため、アジア諸言語の多言語文字表示にも用いられることになったようである。¹⁴⁾

これらの技術では、Web ページ中にフォントを埋め込んでしまうので、Web ページを閲覧するユーザにとっては非常に簡単である。また、画像データを作成するのに比べてデータサイズが小さくなるというメリットもある。ただ、この技術は、比較的新しい Web ブラウザ¹⁵⁾ が必要となる。また、フォントそのものを配布してしまうことになるため、場合によっては著作権上の問題が生じることもある。PDF(Portable Document Format)¹⁶⁾ は、Acrobat Reader さえあればどこでもまったく同じドキュメントを閲覧できることを特徴としており、そのために用意されているフォント埋め込み機能を用いることで、様々な言語を共有することも可能である。PDF はレイアウトを完全に再現できるため、クライアントに依存しないレイアウトを必要とするドキュメントで多言語文字を扱う際にしばしば用いられている。ただ、これらの方法では、a. と同様、今回の CAI プログラムで利用できるような形でのドキュメントを作成するのがあまり簡単でないことから、採用には至らなかった。

c. Java Applet を利用する

Internet Explorer や Netscape では、はやくから JAVA が利用可能になっており、また、Java が Unicode を簡単に利用できることから、クライアント側で何も用意せずに済む多言語文字表示の方法として、Java Applet は 97 年頃にはすでに利用されていたようである。『マルチリンガル Web ガイド』では、97 年当時の Java Applet の利用例として、ペルシャ語、トルコ語、タミル語、中国語、韓国語を挙げている。また、前田亮氏により、受信者側でフォントを用意しなくとも日本語等を表示・入力できるシステムの開発も行われている。¹⁷⁾ また、AA 研においては、フィジー語やチベット語を

¹²⁾ <http://www.microsoft.com/typography/web/embedding/>

¹³⁾ <http://www.truedoc.com/webpages/intro/>

Dynamic Fonts は、Bitstream WebFont Player をインストールすることで、Internet Explorer 4.0 以降でも利用可能である。ただし、Netscape の新バージョンである Netscape6 では Dynamic Fonts をサポートしていない。(http://home.netscape.com/eng/mozilla/ns6/relnotes/6.0.html) Netscape6 の今後のバージョンが Dynamic Fonts を採用していくのかどうか注目する必要がある。

¹⁴⁾ たとえば、ヒンドゥー教寺院の Web サイト、Mata Vaishno Devi 寺院 <http://www.anugraphics.com/vaishnodevi/> では、Dynamic Fonts を利用してデーヴァナーガリーを表示しており、MS-Windows で Netscape Communicator を用いれば、そのままの環境でヒンディー語が表示される。

¹⁵⁾ WEFT は IE ver. 4 以上、Dynamic Fonts は Netscape 4.03 以上、もしくは IE ver. 4 以上となっている。ただし、IE で利用するには新たにソフトをダウンロードする必要がある。

¹⁶⁾ <http://www.adobe.co.jp/products/acrobat/adobe.pdf.html>

¹⁷⁾ 前田 亮, ミリアンダルトア, 太田 純, 阪口 哲男, 杉本 重雄, 田畑 孝「ブラウザ側にフォントを必要としない多言語 HTML 文書の提供システム」第 55 回情報処理学会全国大会講演論文集, 1N-02, pp. 68-69, Sep. <http://db-www.aist-nara.ac.jp/aki-mae/pub/annual55/annual55.html>

前田 亮, Myriam Dartois, 太田 純, 藤田 岳久, 阪口 哲男, 杉本 重雄, 田畑 孝一. クライアントにフォ

WWW 上で扱える Java Applet が同様の形で開発され、WWW 電子辞書として公開されている。¹⁸⁾ チベット語に関しては Wylie 転写方式で入力し、それをチベット文字で Web 上に表示するというものだが、これを応用した Tibetan Writer 2000 という Web 上の入力・表示システムも公開されている。¹⁹⁾ 筆者は、チェンナイ出張中 (2000 年 12 月) にも、Java Applet を利用したタミル語の同種のものを確認した。これは、タミル文字で利用することを目的とした E-mail サーバであり、メールの読み書きのインターフェイスとして用意されている Web メーラーで Java Applet を用いてタミル文字が入力できるようになっていた。タミル文字は様々な文字コードが混在しており、1997 年と 1999 年の 2 度の国際会議を経た上、タミル・ナードゥ州政府の肝いりで、ようやく TAM/TAB と呼ばれる規格が制定され、普及に向かっているという段階である。²⁰⁾ このため、クライアント側にいちいち用意する必要がないこの Web メーラシステムは好評のようだった。

このように、Java Applet は、多言語文字を表示するにあたって大変便利なものである。ただ、現状では、比較的性能の低いコンピュータでの実行速度がかなり遅いという問題があった。このため、Java Applet は今後の課題としつつ、今回の CAI プログラムへの利用は見送ることにした。

d. 文字を画像として表示する

これはもっとも単純な方法である。Internet Explorer でも Netscape でも、画像の表示であれば、かなり初期のバージョンからでも可能であり、画像がそれほど大きくなければ、表示速度もあまり遅くはならない。とはいえ、文字を入力するたびに画像を作ってファイル名をつけてそれにリンクしていくというのでは、やはり、簡単にデータを扱えるとは言い難い。

しかしながら、CGI プログラムを介することで ASCII 文字列を文字画像に置き換える方法という方法がすでに行われており²¹⁾、実行速度的にやや難があったのだが、こ

トを必要としない多言語 HTML 文書ブラウジングシステム. 情報処理学会論文誌, Vol. 39, No. 3, pp. 802-809, Mar. 1998. 他、前田亮氏による博士論文も WWW に公開されている。"Studies on Multilingual Information Processing on the Internet. PhD thesis, Nara Institute of Science and Technology", Sep. 2000. <http://db-www.aist-nara.ac.jp/aki-mae/pub/dthesis-e.pdf>.

¹⁸⁾ それぞれ、
フィジー語・英語・日本語学習用オンライン辞書
(<http://www.aa.tufs.ac.jp/ritsuko/1999/fiji/dictionary.html>)
チベット語辞典編纂室
(<http://www.aa.tufs.ac.jp/hoshi/tibetjiten/jitenframe.html>)

¹⁹⁾ <http://www.aa.tufs.ac.jp/hoshi/dictionary/tibwrpro.html>

²⁰⁾ タミルナードゥ州政府の IT 政策については「IT initiatives in Tamil Nadu」(<http://www.tn.gov.in/its0101.htm>) を参照。なお、TAM/TAB(MONO LINGUAL CODING SCHEME FOR TAMIL / BILINGUAL CODING SCHEME FOR TAMIL) 規格においては、TAM 規格は印刷用途を視野に入れ、母音の文字と子音の文字を結合した形もコードの中に含まれており、TAB 規格は、ASCII 文字との共存を考慮したものである。詳細については、<http://www.tamilnet99.org/tabtam.htm> を参照。

²¹⁾ ペンシルヴァニア大学の Linguistic Data Consortium では、ASCII 文字を入力すると IPA フォントを GIF 画像で提供してくれる IPA-GIF Server (<http://www ldc.upenn.edu/cgi-bin/sb/ipagif/ipagif.cgi>) がすでに公開されており、また、東北大学情報科学研究科の相場徹氏による「Play e 漢字 ページ」も、サーバに漢字番号等を送信することで、京都大学人文科学研究所の e 漢字フォントを jpg や gif の画像に変換して利用できるサービスを提供していた。

の種の、サーバ側で画像を生成するシステムの場合、文字表示のための実行速度はクライアント側ではなくサーバ側に多くを依存することになるため、画像生成の速度の問題さえ解決できれば現状ではベストな選択と言えそうだった。そこで、画像生成速度が可能な限りはやくなるような形で、アスキーに転写した文字を WWW サーバに送信するとその文字の画像を返してくる CGI プログラムの作成を試みることにした。

3 アジア諸語語学教育 CAI プログラムのための文字表示システムの開発

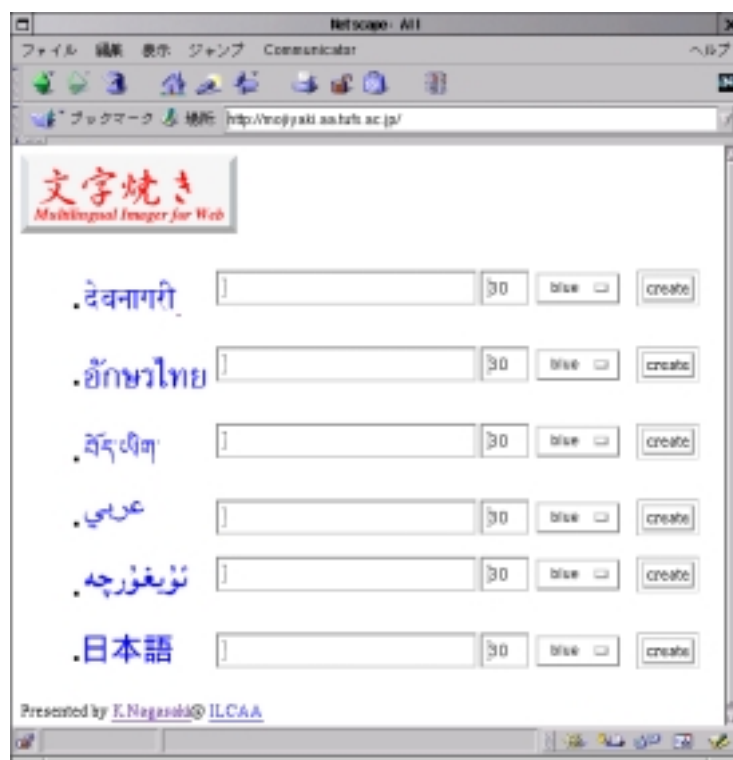


図 1: 「文字焼き」(多言語文字画像表示システム)

上に述べたように、このシステム²²⁾は、技術そのものは実現可能であることがすでにわかっていたため、既存の技術を利用していかに使いやすくかつ速いものにするかというところに重点をおいて開発を行った。

²²⁾ このシステムは、「文字焼き」と名付けた。この名前は、AA 研 COE 研究員（当時）小林正人氏により発案されたものである。

3.1 システムの構成

文字を画像として表示させるシステムは、主に、以下のフリーソフトを利用した Perl スクリプトによる CGI プログラムとして作成した。

- Vine Linux2.0CR ²³⁾
- Apache ²⁴⁾
- Perl ²⁵⁾
- mod_perl ²⁶⁾
- ImageMagick ²⁷⁾

中でも ImageMagick は、Truetype フォントと Unicode を引数で与えると該当する文字を画像表示してくれるという機能を持っており、また、Perl のモジュールとしても動作するようになっていたため、今回のシステムには大変有効であった。mod_perl は、perl の CGI プロセスを Web サーバソフトのプロセスに組み込んでしまうことができるので、CGI による文字画像生成の高速化に大変役立った。

3.2 各文字への対応

対応している言語（文字）は、2000 年 6 月以来、徐々に増え、現在は以下の通りである。

- デーヴァナーガリー
- タイ文字
- チベット文字
- アラビア文字（アラビア語、ウイグル語）
- 日本語

以下に個々の版について解説する。

²³⁾ <http://www.vinelinux.org/>

²⁴⁾ <http://www.apache.org/>

²⁵⁾ <http://www.cpan.org/>

²⁶⁾ <http://perl.apache.org/>

²⁷⁾ <http://www.imagemagick.org/>

क+ष = क्ष

図 3: k + sha = ksh

かでそれを置き換える処理を行わないと正しい表示にならない。この場合にも、適切な形でテキストを表示してくれるシステムが必要となってしまうのである。

今回のスクリプトでは、文字が細かい部分に別れているフォントを用い、組み合わせながら表示する方法を採った。入力された文字列を表示したい配列に変換してから Unicode の 16 進数文字列に変換し、それを ImageMagick の Perl のモジュールに与えてやることでデーヴァナーガリーを表示できるようにした。ただ、フォントのコード割り当て自体は ASCII 文字列のコードを中心にしたものであり、また、リガチャについてもフォントで使っていない余ったコードを割り当てているだけだったため、実際の Unicode のデーヴァナーガリーと互換性を持たせることはできなかった。

先述したタミル文字では、こうした問題に対処できるように TAM/TAB という体系が開発されたのだが、タミル文字に比べ、デーヴァナーガリーのリガチャは比較にならないほど多いため、リガチャーを文字コードに含めるのは一大作業となってしまう。ということから、フリーのフォントの多くは、リガチャーを作りやすい部品を提供するという形になっているものが多い。今回採用したものは部品を組み合わせで文字にするタイプのものであった。³¹⁾

3.2.2 タイ文字版

タイ文字版³²⁾は、東京外大 AA 研峰岸真琴氏が考案し、Thai TeX で用いられている翻字体系に基づいて作成した。タイ文字には、デーヴァナーガリーと同様の、母音と子音の前後関係が逆転するという現象があるが、これに関しては、Thai TeX 用のスクリプトの一部を借用することで解決した。³³⁾

3.2.3 チベット文字版

チベット文字版は Wylie 転写方式を用いた。³⁴⁾チベット文字は前後が入れ代わるということはないが、基字に対して、有足字、有頭字、母音符号等といった形で上下に様々な文字が追加されていくため、場合によっては、基字の形が変わってしまうことがある。そのため、今回のシステムでは結合した形もフォントとして用意しておく必要が生じ、幾つかの結合文字を作成することになった。とはいえ、仏教経典等にみら

³¹⁾ インドの言語のフォントについては <http://jeff.cs.mcgill.ca/luc/indic.html> ここに詳しい。また、ネット上で入手可能なフォント全般については <http://jeff.cs.mcgill.ca/luc/fonts.html> こちらを参照。

³²⁾ これに関しては、東京外大 AA 研峰岸真琴氏に貴重なアドバイスをいただいた。

³³⁾ 筆者が借用した Thai TeX は、東京外大 AA 研高島淳氏が作成されたものである。

³⁴⁾ チベット文字版に関しては、東京外大 AA 研星泉氏に貴重なアドバイスをいただいた。

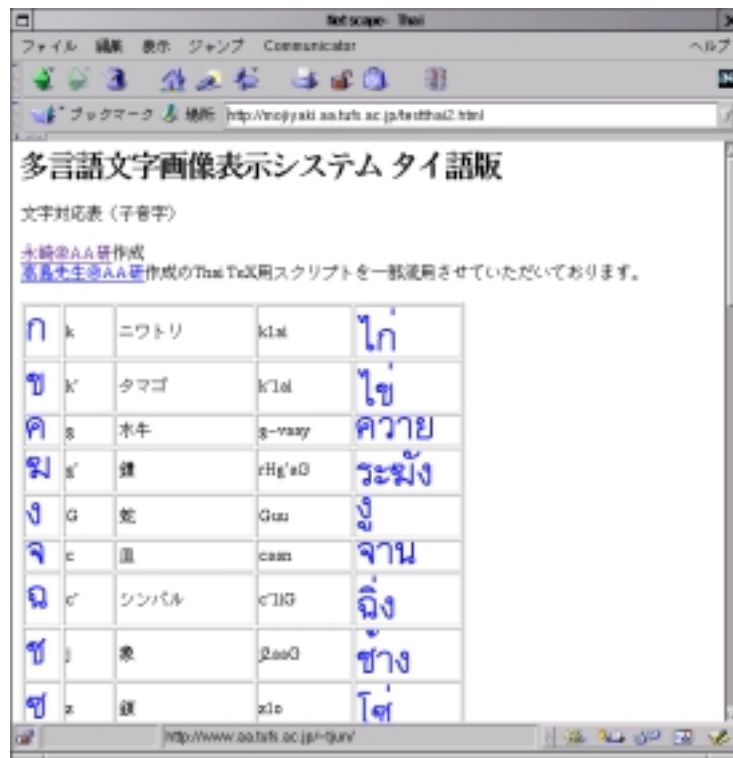


図 4: 「文字焼き」多言語文字画像表示システム タイ語版)

れるサンスクリットの音写については様々な変則的なパターンがあり得るため、そうしたものの対応は今後の課題とした。

3.2.4 アラビア文字（アラビア語版、ウイグル語版）

アラビア語、ウイグル語版に関しては、³⁵⁾ Arab TeX の転写方式を用いている。アラビア系文字はほとんどの文字に Unicode が割り当てられているため、インド系文字のように別にリガチャを用意する必要がなく、その点では比較的作成は用意であった。ただ、アラビア語とウイグル語では表記のシステムが異なり、前者は一定の規則にしたがって文字が連結するのに対し、後者は基本的に文字が音節ごとに離れることになっているため、使用している文字は同じでも、システムを別に作る必要があった。

3.2.5 日本語版

日本語は、英語版等の MS-Windows でもインターネット上で無償で配布されている Global IME をインストールすればほぼ完全に扱えるので、日本語を入力することでそ

³⁵⁾ アラビア語に関しては AA 研 COE 研究員榮谷温子氏、ウイグル語に関しては、東京外大 AA 研 COE 研究員菅原純氏に再三にわたる貴重なアドバイスをいただいた。

れを用いるという前提で作成した。³⁶⁾

3.3 まとめ

このシステムを利用することで、アジア諸言語語学 CAI プログラムは、HTML のフォームに ASCII 文字を入力するだけで必要な文字を含んだ問題を作成できるようになった。

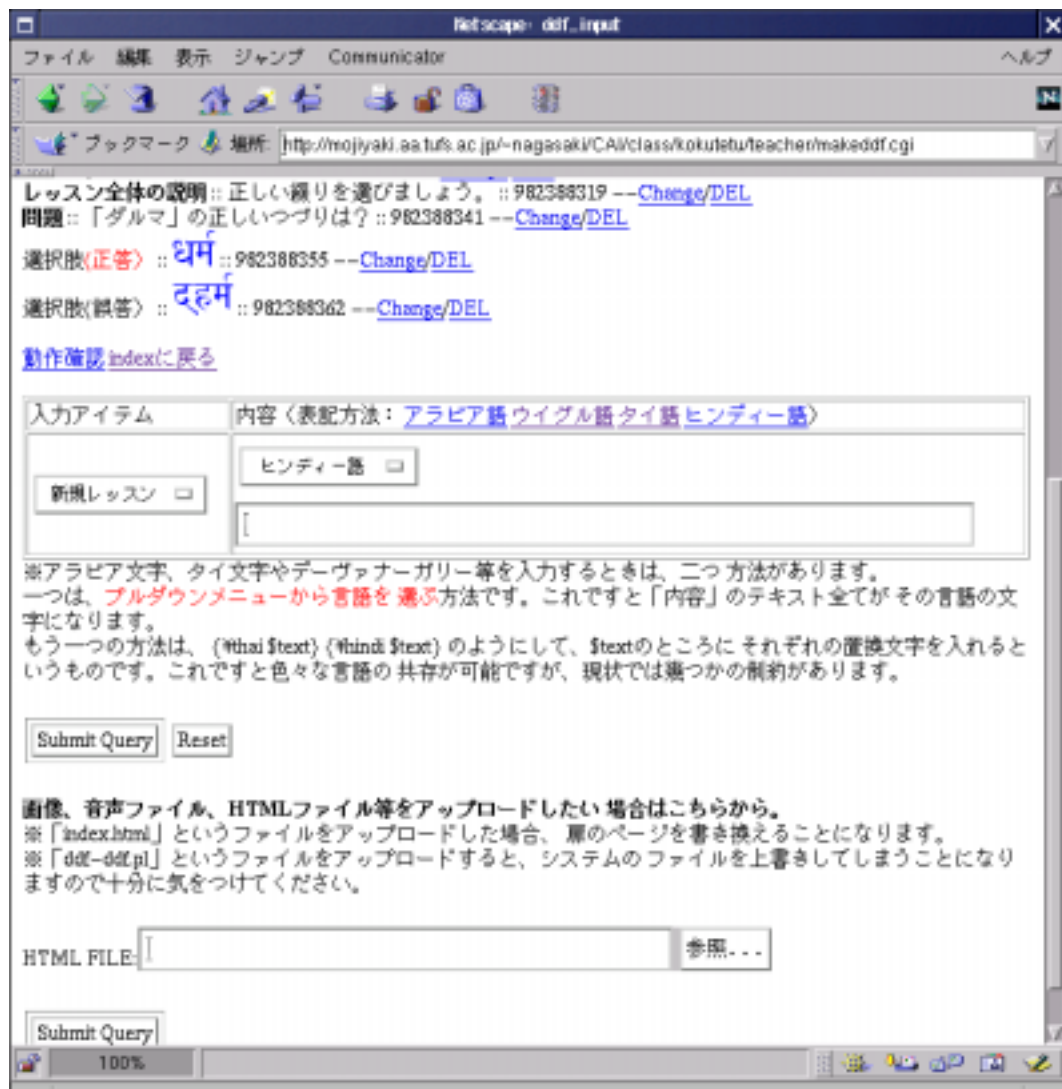


図 5: CAI プログラムの問題作成システムの一部

現在、このシステムを用いた CAI プログラムとして利用可能なものには、榮谷氏によるアラビア語 CAI がある。³⁷⁾

³⁶⁾ 日本語スクリプト作成に際しては、perl5.6 の utf8 機能及び Jcode.pm を利用した。

³⁷⁾ <http://mojiyaki.aa.tufs.ac.jp/~nagasaki/CAI/class/arabiyyah/>

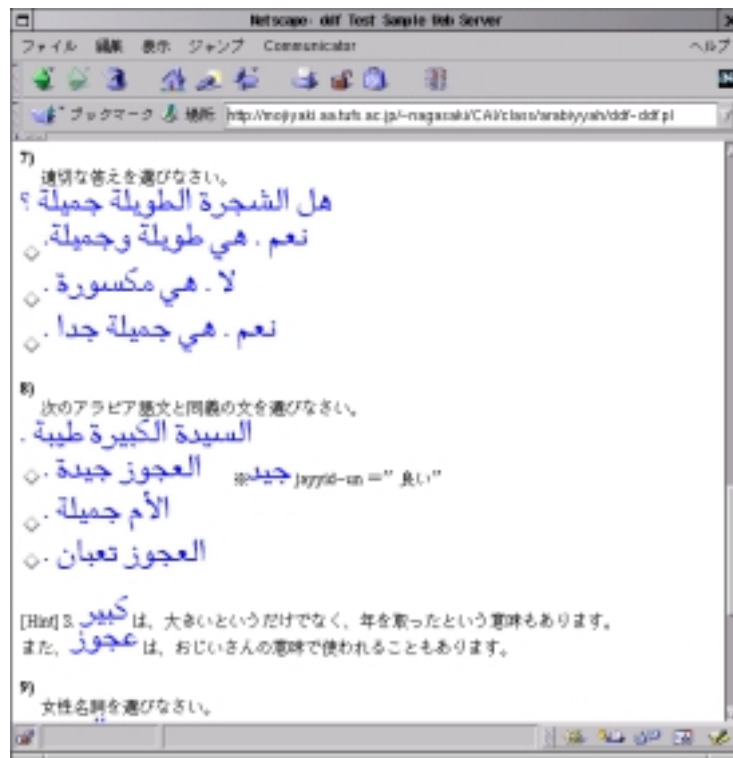


図 6: 榮谷氏によるアラビア語語学 CAI の一部

また、このシステムは、CGI プログラムに GET Method を用いたことで、IMG タグを用いて HTML 文書中に気軽に埋め込むこともできるようになっている。たとえば、

```

```

このような書式で sort=の値に ASCII 転写した文字列を入力すれば、当該文字列（この場合はデーヴァナーガリー）の画像に変換されて HTML 文書の中に表示されることになる。この程度であれば、HTML の書き方を多少覚えるだけで利用可能である。

さらに、この機能は、CAI プログラムのみならず、他の CGI プログラムから呼び出して利用するのにも適しており、Web 上の電子辞書の検索結果の表示等といった比較的短いテキストの画像表示に有効である。現在では、たとえば、チベット語版に関しては東京外大 AA 研町田研究室のヒンディー語電子辞書や³⁸⁾の星研究室のチベット語電子辞典の³⁹⁾や検索結果表示に用いられている。

繰り返しになるが、このシステムは、技術としては決して目新しいものではない。将来的な方向性としては、Dynamic Fonts 等の Font embedding の利用可能な環境の割合が一層高まっていくであろうことや、プロセッサの高速化やネットワークの広帯域化していくであろうこと等を考えるなら、今後は Dynamic Fonts や Java Applet といっ

³⁸⁾ http://www.aa.tufs.ac.jp/~kmach/hnd_la.htm

³⁹⁾ <http://www.aa.tufs.ac.jp/~hoshi/tibetjiten/jitenframe.html>

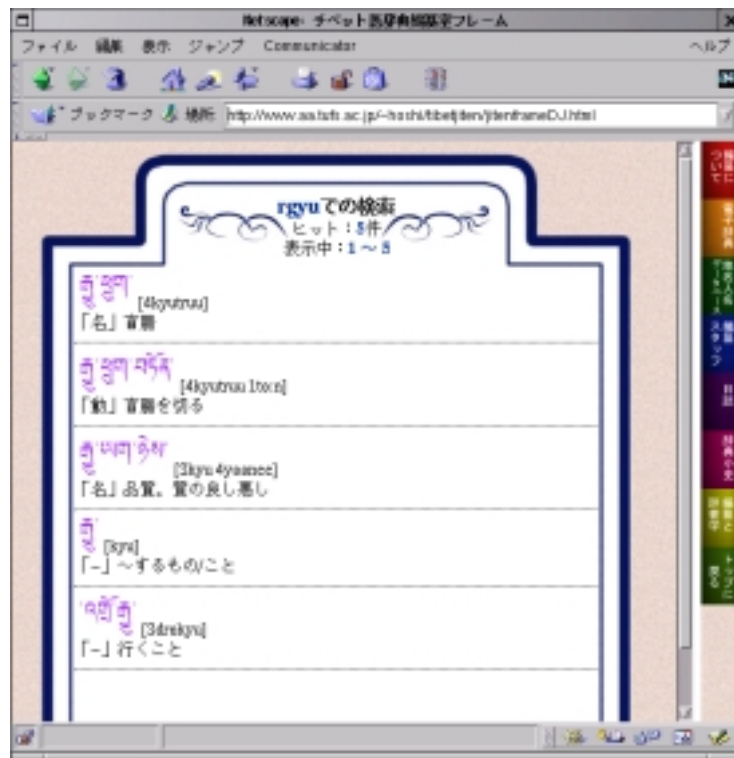


図 7: 星研究室チベット語電子辞典の検索結果表示画面

た方法が主流となっていくであろう。しかし、現状のインターネットにおいては、可能な限り幅広く対処できるものである。2001年3月になって、パーソナルメディアが「超漢字ウェブサーバ⁴⁰⁾」として、超漢字 OS 以外の環境に対しては多漢字、多文字部分の文字フォントを PNG 画像で文字を出力するというシステムをリリースしたのは、文字を画像で表示するシステムがまだ有効性を失ったわけではないという一例と言えるかもしれない。⁴¹⁾

*なお、参考文献については各注の中で記しているので、そちらを参照されたい。

⁴⁰⁾ 17 万字超の文字を Web で表示可能であるとしている。http://www.chokanji.com/

⁴¹⁾ なお、筆者はまだ「超漢字ウェブサーバ」を利用した経験がないので、このソフトウェアにおける実際のシステムの有効性についてはコメントできない。早急に試用してみようことを検討中である。