Designing Methodologies of Tamil Language in Web Services

C. Santhosh kumar[1] , D. Palanikkumar [2], P. Seethalakshmi[3]
[1]PG Scholar, Dept. of CSE,
[2]Asst.Professor, Dept. of CSE,
[3]PG Scholar, Dept of IT,
Anna University of Technology,
Coimbatore,India
cjsksag@gmail.com

Abstract—**Nowadays most of the effective conversations in the internet services are in English language. Tamil computing, a latest emerging technology which has a method of using Tamil scripts in internet services. This paper suggests the theoretical approach of designing interfaces, methods applications, and database usage in Tamil computing.**

Keywords— Designing Interfaces, Database Usage.

I.   INTRODUCTION

Computing with Tamil relates to the design and development of useful applications which permit interaction with the system in Tamil. The applications could be standalone applications running on different platforms or web based applications such as search engines, instant messaging or chat. Tamil consists of twelve vowels and eighteen consonants. Grammar rules are quite specific, reading out Tamil will pose some initial difficulties. The essential idea here is that people will be able to use computers to deal with information given in Tamil. This means that input and display are both supported in Tamil and will not require the use of English. Any system of computing has to meet the basic requirement that it be able to accept process and display text based information. Ultimately, it is text that relates directly to human communications and hence it is quite important that a computer supporting Tamil based interfaces should in the least, be able to handle data entry and display in Tamil, if not text processing. Electronic information processing is not critical for displaying and printing the text since very little processing is involved on the entered data except formatting. The techniques available for doing this with English may be easily adapted to handle Tamil. However true computing involves the collection of varied items of information which are pooled together and processed to produce results based on the user requirements, such as search for information in a data base. Applications involving records in public institutions are examples of text processing on a large scale. Information Technology, in respect of usage of Tamil on computers on a State-wide scale, should necessarily aim at reaching all the people so as to give them the benefits of this new technological wonder. This is not a simple matter that can be handled by designing fonts or standardizing keyboard

mappings. The solutions will not provide a simple, easy and uniform way of communication that can be used by all people. It is necessary to look at the problem from the angle of information processing for the long term growth of the Tamil language both for electronic processing and the socio-economic progress of the state. In a wider context, Tamilnadu should be able to share its expertise to provide IT solutions for the rest of the country in their respective languages. Software and business opportunities throughout the country will then come within the reach of the Tamilnadu software industry. We should not loose the opportunity in proposing a viable approach to Information technology in Indian languages. Initially I explained about the design approaches, later about the methods, coding schemes and finally about the applications.

II.   LITERATURE WORK

At this conference [1], it was proposed that the placement of glyphs within a Tamil font would follow a recommended scheme. Both bilingual and monolingual schemes were standardized. In the bilingual scheme, the initial 96 character positions are retained for Roman letters while the top 128 are allocated for Tamil.[2] In the monolingual scheme, all the glyphs are used for Tamil. The conference solves the standard for data entry in Tamil, Three different keyboard layouts were arrived at for use by different sections of the users. The first relates to what is termed as the phonetic keyboard where data entry is affected through lower case keys alone for the basic text. the second scheme referred to as the Romanized keyboard, specifies data entry based on the Roman letter that comes close to the sound of the vowel or consonant. The third is the layout seen in standard Tamil typewriters. A look at the approach taken by IIT Madras to synthesize speech in Tamil as well as other Indian Languages. The results are

extremely satisfying, despite a robotic flavour to the synthesized output.[3][4] The speech enhanced Multilingual Editor and other applications have found acceptance by the Visually Handicapped community in Tamilnadu as well as other states in India. We believe this to be the VERY FIRST demonstration of continuous text to speech generation on the web in any Indian language. The attempts at the center for the development of advanced computing have resulted in some standardization in respect of coding and the internal representations of text in Indian languages.[7] This approach, which became popular through the GIST technology, has fairly severe limitations in practical use especially for Tamil. The CDAC approach does however allow a number of DOS based applications to work with Indian scripts. There are many difficulties in using the system because of the restriction on the platform and the hardware approach in using the GIST card. The ISCII coding scheme used by the software does allow some computing in Indian languages.

III.   TAMIL COMPUTING - A THEORETICAL APPROACH

Fig. 1.1. Tamil Computing problems and Rectifications

For many years now, text in Tamil has been displayed on the web, thanks to the magazines which have gone on-line. Each publication has standardized the approach to displaying the text through designated fonts and there are quite a few of them. Besides the magazines, independent groups have proposed data entry schemes to be used with specific fonts which seem to cater to some sort of character coding for the characters. The multiplicity of the fonts seen has posed real problems in arriving at some uniformity in text display. The proliferation of fonts seems to continue as also specific encoding recommendations. Most of the efforts seem to relate to the Win9X platforms. The new TISCII recommendation seems to be gaining some ground, as seen from the increasing reference to it in the web. Unicode has become a world standard and many computer applications have provided Unicode support so that multilingual text can be handled. The Unicode standard proposed for Tamil has not taken into consideration some of the important linguistic issues. Also, even among the professionals, there seems to be considerable difference of opinion in respect of the adequacy of Unicode. At the Systems Development Laboratory, the view held is that Unicode is not really suited for text processing in Indian languages though the data entry and display requirements could be handled with the current Unicode assignments for Indian languages. There are differing views about the

suitability of Unicode, even in respect of Tamil. The IITM software, on account of its flexible approach to computing with Indian languages, was able to support the requirements specified in the Tamilnet99 standards. The multilingual editor conforming to the Tamilnet99 standards was easily developed at the lab and has been made available for general use. The real power of the IITM software becomes apparent in applications that require linguistic analysis of text in Tamil. The links below refer to many applications that have been developed at IIT Madras for document preparation and linguistic processing with Tamil. A minimal computing environment should therefore support the concept of internal representation of information in Tamil in a meaningful way as to permit a computer program to interpret the information and process the same. Thus there are three essential components to a computing system in Tamil.
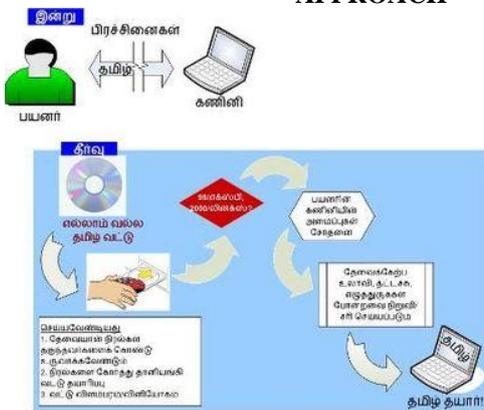
1. Data entry of text in Tamil.
2. Display or printing of text in Tamil.
3. Processing of the text to suit user requirements.

Approaches to dealing with all the three have been standardized for the western languages. This standardization has been effected through the concept of character codes and fonts. This standardization has allowed English to be handled in a uniform manner on all computers which use the standard ASCII code for all text processing. It turns out that data entry, display or printing of Information in Tamil can be effected using the standardization accepted for English, by designing suitable fonts for the display of Tamil letters and using conventional word processing software to effect data entry. This way, word processing and Desk Top Publishing programs are fooled into believing that they are handling Roman text (English). Thus, with a suitable font and the use of the Macro facilities of word processors which allow a sequence of key strokes to be interpreted together, data entry and printing in Tamil may be comfortably handled. Many commercial packages take this approach to providing DTP facilities in Tamil. Unfortunately, such solutions cannot provide meaningful support to the real text processing requirements in Tamil which has to be handled purely from a linguistic or language related point of view. To perform any meaningful text processing in Tamil (or for that matter any Indian language) one has to work with the syllabic structure of the language and not merely with the vowels and consonants. This is a fairly complex issue as it involves individual identification of hundreds of characters which result through the combinations of the basic vowels and consonants. The phonetic aspect of Tamil makes it necessary for one to relate the written form to the linguistic units and there lies the basic difficulty in deriving a suitable internal representation for the text. A number of available approaches to dealing with Tamil on the computer have not looked at the text processing requirement with much seriousness, and have concentrated primarily on data entry and printing only. [8]As a consequence, one sees a multitude of fonts for Tamil, each with its own accompanying data entry software. One need not be alarmed at this as most of these offer working solutions capable of producing quality printouts or web displays on the internet. The need to look at standardization arises only when one considers the issue of text processing. While

such processing is in principle provided for with the internal representations used with fonts, it is nevertheless a complicated task as it bears no relationship to the linguistic aspects of Tamil. To summarize, computing with Tamil should address the problem of text processing. In the western world, the writing systems are based on a small set of symbols and letter shapes. The software caters to the standardized representations of the text through the ASCII code. The letters of Tamil do not lend themselves to such interpretations, for they represent sounds at the level of a syllable derived from a basic set of vowels and consonants. Each syllable is unique from a linguistic point of view and any effort at standardization or electronic representation of text should necessarily conform to linguistic requirements. A simple but very effective multilingual text editor conforming to the syllabic requirements for linguistic processing. This editor provides support for generating displays with a variety of fonts, including the fonts such as Tamilnet99, [5] Anjal, Murasu, Mylai, Tiscii and more. This editor also provides variations in internal storage for the letters which have the same shape but differing sounds. The data entry scheme for the editor is flexible.The set of utilities developed for linguistic processing in different Indian languages includes a program for computing the frequencies of occurrences of vowels, consonants and their combinations in any given text. Essentially the program does a count of the syllables and tabulates the results in a useful manner. The link above will also take you to the results of frequency counts of the aksharas in Tirukkural and Sambhandar Tevaram. Lexical ordering according to the specified order of the letters of Tamil has been a major issue and this specific problem has not been given sufficient attention by those developing standards for Tamil. The Unicode assignment for Tamil is a hopelessly mangled set of the letters but the claim is that Unicode does not purport to preserve lexical ordering. The IITM Software preserves the lexical ordering and utilities for sorting and indexing text prepared using the Multilingual editor. It is also possible to write utilities in PERL to effect text processing and regular expression matching with text in indian languages. The section on PERL modules for Indian languages has additional information. There is also the fundamental question about what constitutes a proper sorting order for Tamil. This question can be answered only after the full set of aksharas and special symbols required for regular use are correctly identified and codes assigned for the set. An example of a web based application searching for words in the text of Tirukkural, Tolkappiam and other works in Tamil. This is a Java based web interface and allows data entry of text strings in Tamil. This is a unique presentation.

IV. DESIGNING USER INTERFACE

1) Characters used in Tamil

There are twelve basic vowels and eighteen consonants in Tamil. The consonants are divided into three groups. The "Ayda" letter is often viewed as the thirteenth vowel. Each consonant in its intrinsic form is given without any vowel. Often in Tamil primers the student is taught the consonants with the vowel "a" since it is easier to learn the sound associated with the consonant this way. Hence the "Ayda" letter is also imagined to be a vowel which when combined

with a consonant results in its intrinsic form. Thus there are 13+18+(18*13) different letters or aksharas which are significant linguistically. [9] Tamil has its own representation for the numerals and there are other special symbols to represent the date, month and the year. Given below is the set of symbols which should be considered basic to the script.

Fig. 1.2. Type of characters in Tamil

Fig. 1.3. Standard Character set for Tamil in ISO 8859-x

Added to the eighteen consonants are six consonants from Sanskrit which are used fairly regularly these days, though one does not see them in ancient Tamil texts. The thirteen combinations of each of these with the basic vowels is also to be represented. Tamil has its own representation for the numerals. While writing, a consonant vowel combination is not always subject to the same rule of adding a vowel extension. Special ligatures are used to represent consonant vowel combinations. There are rules for writing a consonant-vowel combination but the rules do not apply uniformly for all the vowels.

Though punctuation marks are rarely seen in older texts, there is a need to use some of the special symbols and punctuation marks if we have to use the language for preparing educational material and scientific text.

Though Tamil comes under the general category of phonetic languages, the pronunciation of some intermediate letters in words is context dependent and from a linguistic angle quite important. Hence some mechanism to distinguish the different sounds produced by the same letter will be helpful in the preparation of educational material for teaching Tamil. Lastly, the writing system for Tamil has evolved over a number of centuries and it is necessary to display ancient tamil text in the same scripts used in old manuscripts and stone inscript.

2) Method of using Tamil fonts

Many of the solutions provided in the past for dealing with Tamil on computers have relied on the use of special fonts. A font is essentially a set of basic shapes which may be shown individually or in combinations to display the

letters of a script. Eight bit fonts are the most commonly used fonts with Indian scripts. Seen below is the basic shapes (called Glyphs) which permit Tamil text to be displayed on a computer screen, as present in the font known as Indo-Web-Kambar.

Fig. 1.4. Tamil Fonts

To understand how fonts are used, consider the string shown below which string is built up from individual Glyphs.

Fig.1.5. Font Usage

There are 10 shapes here (each is known as a glyph) but only seven of them are distinct. [10]The glyphs within a font are arranged in some order and inside the computer, the shapes to be displayed are given in terms of the positions occupied by the glyphs in the fonts. Typically a font will support up to about 200 glyphs. In the Kambar font used in displaying the Tamil string above, the glyphs used in the string are (234, 235, 168, 200, 208, 226, 235, 232, 200, 168) in the order in which they will be displayed. It may be noted that the string has only 5 Tamil letters though 10 glyphs are used, 7 being distinct. Thus going by the number of Glyphs alone, one will not be able to figure out the number of letters in the string. In other words, linguistic analysis of text will be cumbersome if font based representation is chosen. The method most often used for representing information in Tamil is based on glyph codes for the letters. This works alright for a given font and to some extent string processing may also be attempted. However the dependence of the codes on the font is a major deterrent to using this approach. The glyphs required to display a string are generated when the keystrokes are effected. For this, the keys on the ascii keyboard are mapped into the tamil letter whose glyph is specified through the ascii code for the letter. Some combinations will require two keystrokes but this is alright since the second keystroke will correspond to a vowel extension. Inspite of its simplicity, this method is quite painful in practice, for it is always not easy to remember the mapping between the Tamil letter and the key. Worse still, some combinations require three keystrokes to be executed whereas other combinations may be handled through just one keystroke. An example of this is the difference between keyboard entries for "ti" and "tu". Depending on the font design, the number of glyphs and therefore the number of keystrokes will vary. This is not a useful approach for general acceptance. It is thus apparent that Tamil cannot be typed in just as English is. Somehow the

combinations have to be handled in a font independent manner with a uniformly same number of keystrokes for combinations. Some word processors provide special support by tracking the keystrokes and combining them appropriately. Packages from CDAC or word processors such as SRILIPI use this method. Here too, the sequence is decided by the glyph mappings though in the CDAC software, a special mapping known as the INSCRIPT is used but this is not intuitive for those familiar with English. With most software, one has very little choice for the fonts since the data entry method is in some way tied to the use of fonts. No two font designers agree on the glyphs nor their placement within the font. Worse still, fonts are not always compatible across computer systems. Looking at the problem of keyboard mapping itself, the key to be pressed for a specific letter is fixed by the glyph code. This key may have no phonetic equivalence with the Tamil letter. In fact for many fonts where the designer had intended bilingual use (Roman and Tamil), the glyphs for Tamil are located in the 128-255 range making data entry even more difficult, unless the MACRO features are invoked.One solution recommended by some designers has been to place the glyphs of the Tamil letters at positions corresponding to the ASCII code of the phonetically equivalent letter in Roman. This makes data entry a bit more intuitive but here too variations occur when dealing with consonant vowel combinations which change the basic shape of the consonant.[7] In the scheme followed at CDAC, internal storage is not in terms of glyph codes but correspond to the ISCII scheme. The special word processor transforms the keystrokes into appropriate letters and an output module converts the ISCII based internal representation into glyph codes. This method has the advantage that it applies to all the Indian languages. But the scheme itself suffers from some language specific representations. The ISCII code is an eight bit code that codes only the basic consonants and vowels of the language. Consequently it requires more than one byte to represent a combination though a consonant or a vowel by itself requires only one byte. Thus ISCII also amounts to a multibyte variable length code making the font rendering mechanism quite complex. Nevertheless ISCII is a code that represents basic sounds and hence is quite useful in practice. In implementation however, ISCII has run into some problems for South Indian scripts, especially Tamil.

3) Basic methods for coding scheme
1. Codes must necessarily correspond to syllables which form the linguistic base for the language. Also the coding must use fixed length codes, even if multibyte. This is the best way to handle complex string processing issues consistent with the phonetic nature of our languages.
2. It will be helpful if glyph positioning within a font for Tamil have some relationship with the internally assigned codes. Such a provision will help in string processing.
3. Codes assigned must conform to the dictionary sorting order for the letters of the language.
4. Codes must also be assigned for the special symbols used in the language, numerals included.
5. Codes should identify the consonant and vowel forming the syllable. This will help in linguistic analysis as well as conversion to other formats. This will also help in speech synthesis.

6. Coding should take into account syllables found in other languages of India so that transliteration into Tamil is easily effected. This will help teach other languages through Tamil.

7. The assignment of codes may also take into consideration the numeric values traditionally assigned to the letters of Indian languages.

8. Codes should have no relationship to the glyphs used to display the letter. This is essential to make sure that the internal representation is independent of the font rendering process. Only then we have the possibility of using the software on different platforms. This recommendation does not really run contrary to the observation made earlier that glyph positioning may be influenced by the internal representation.

4) The design of fonts

1. It will be useful to fix the glyphs in the displayable range for ASCII since data entry can be effected from wordprocessors. This is feasible for Tamil. This way editing can be done on the text string using conventional wordprocessors. Even if placed in the 128-255 range, the glyphs may be located at positions corresponding to the equivalent ASCII in the lower range.

2. It is a good idea to offer a minimum set of glyphs for each script and fix their locations as well. This will make it easier to view the same text in different fonts so long as the text does not incorporate special symbols.

3. Font rendering methods vary across systems and it is a good idea to build fonts with standard encodings so that they may be rendered on different platforms using the same glyph codes. This specification is important for web based applications. The default encoding prefered for Tamil is the latin-8859-1 encoding which is well supported on all the three platforms. Also, the minimal set of glyphs may be used on all the fonts with some advantage.

4. It is a good idea to include punctuation and special symbols in the glyphs for the language. This way it will be possible to treat special symbols as part of the language itself. Punctuation should include the period, the comma, exclamation mark and the question mark.

5. It will be helpful to provide some special symbols used in ancient manuscripts of Tamil (e.g., shapes for some double consonants).

6. It is inevitable that variable width characters be designed. In the glyphs, it is recommended that the width is made a multiple of some basic unit (say 2pixels). This will help retain vertical alignment in the text without having to resort to special formatting. One glyph may be retained as a special space whose width is an odd number of units. This glyph will be useful in retaining the alignment.

5) The use of keyboard

It turns out that key mappings may be assigned arbitrarily and the processing software can do the required mapping to the internal codes. Though in earlier sections we hinted at mappings that bore some relationship to the font glyphs, it is useful to look at keyboard mappings from the user point of view rather than programming convenience. The following are some of the recommendations.

1. Data entry should be natural and must relate to the letters of Tamil. It should be easy enough to train persons in the use of the standard QWERTY keyboard which is what one will see on all computers.

2. As far as possible, use only the common keys to map the letters. Not all the keyboards will have all the keys seen in the PC keyboards.

3. Standardize the input method for entering a consonant vowel combination.

4. The manual typewriter keyboard (for Tamil) is a choice that should not be ignored. It is an existing standard and many have been trained in it. This keyboard is entirely adequate for modern Tamil writing. The processing software can always use a keyboard filter to transform the sequence of keystrokes into appropriate letters. This may be accomplished through the use of Macros in most word processors.

5. Do not fix or relate key mapping to any fonts though this is an approach that may allow virtually any word processor to handle data entry in Tamil. What is desirable is that the internal representation be exportable to some word processors so that the powerful formatting facilities seen in them may be utilized.

6. Keyboard mappings should not also relate to the codes assigned for the characters. This is to ensure that the internal representation is independent of the system in which we are processing the text. It is always easier to work with standard codes that do not directly relate to any hardware specific aspect of a computer. Tables can be used to relate the internal representation to the glyphs or keyboard mappings thus allowing great flexibility in dealing with the input and the display.

7. Keyboard mappings arrived at on te basis of some studies on the observed frequencies of the letters in normal Tamil writing are no doubt helpful. It turns out that writing styles vary so much that the frequencies seen in ancient texts are quite different from the frequencies in modern texts. It is therefore preferable to look at the thinking process as one types and assign the key mappings based on user recommendations.

V.   APPLICATIONS

[6]Computing in Tamil or with Tamil has to necessarily provide support for some basic applications that will help bring information technology closer to the people. While one might wish to have virtually every popular application run in Tamil, we need to view some applications as being important to begin with. Given below is a list of applications that should be made available in the vernacular at the earliest. Whatever standardization is contemplated in respect of representing Tamil inside the computer, the recommendations should keep these applications in mind and offer viable technical solutions.

1. Data preparation applications. Data entry and printing applications which may include DTP, both for general use and commercial publishing. Large scale newspaper and magazine publishing as well as archiving. Display and dissemination of information through the web. The applications include tools for preparing HTML documents, Search engines, Archiving software, Applications for preserving ancient manuscripts in their original scripts. Generation of on-line references such as dictionaries which may be queried

2. Text processing applications. Applications involving sorting, indexing and searching of large volumes of data. Data base systems supporting interaction in Tamil. Report generation should be supported in Tamil as well. This will help maintain records in many Government, public and private institutions. The system should supported on standard Data base packages such as Oracle, Access, Informix etc.

3. A text to speech enhanced version of the editor has been provided for the benefit of the Visually Handicapped.

4. Sending and receiving email with text in Tamil or handling chat has been greatly simplified. All that is required is a cut and paste into the application, Outlook Express, Instant messenger or similar ones, from text entered into the Multilingual Editor. Using the Multilingual Editor, email and chat are just one simple paste operation.

VI. CONCLUSION

The Tamil computing shows the way for using the native language in internet applications. The ordinary person without knowing English language will understand and use the applications. The research work in Tamil computing will make the application more user perspective and make the access in all levels of the peoples in the world.

REFERENCES

[1]http://acharya.iitm.ac.in/tamil/tamilcomp.php
[2]1997 at a conference in Singapore called the First Tamil Internet Conference.
[3]The Second Tamil Internet Conference held in Chennai in February 1999.
[4]The 1999 Chennai conference, Tamil Nadu got a Tamil Virtual University and the Tamil Software Development Fund.
[5] www.google.com/images/Tamil computing
[6]http://srinix.wordpress.com/2007/08/29/tutorial-how-to-store-utf8-indian-language-data-in-mysql/
[7]http://www.hotfrog.in/Companies/C-DAC-Gist-PACE-Multilingual-Computer-Training
[8] http://acharya.iitm.ac.in/tamil/desirable_set.php
[9]http://en.wikipedia.org/wiki/Tamil_language
[10]http://en.wikipedia.org/wiki/Tamil_literature
[11]http://en.wikipedia.org/wiki/Tamil_people