International Symposium for Tamil Information Processing and Resources on the Internet

*

Jointly organised by the National University of Singapore and
the National Institute of Education of Nanyang Technological University
on 17 May 1997 at the Auditorium, Computer Centre, NUS.

*

17-18 May 1997, Singapore

Conference Chair - Naa Govindasamy, National Institute of Education &
Prof. Tan Tin Wee, Internet Research Centre, National University of Singapore

Theme/Topics

The theme of the symposium was to promote the use of Tamil on the Internet. Presentations were geared towards a wide range of audience from general Internet users to content developers.

Suggested topics of interest included, but were not limited to:

-- Tamil and the Internet
-- Tamil and the World Wide Web (WWW)
-- Character set issues on the Internet and WWW
-- Standardization
-- Keyboard Input Methods
-- Language processing issues with Tamil
-- Education on the Internet
-- Library and archival concerns
-- Business-related topics
- Case studies

Sessions

The two day Symposium provided the following type of sessions:

— Keynote presentations
— Technical presentations
— Open discussion sessions
— Discussion Forum - A Summary Report

The 2 hour-long discussion session on the second day of the symposium came to a fruitful ending. The discussion panel (a total of 11 members) was led by Sujatha and included N. Anbarasan, S Kuppuswami, ST Nandasara, Kuppuswamy Kalyanasundaram, Malaan, Harold Schiffman, Vasu Renganathan, Muthelilan Murasu Nedumaran, Naa Govindasamy and Tan Tin Wee. The topics discussed includes -

— Increasing Tamil Content Worldwide
— Keyboard Diversity
— Encoding Standards
— Standardization of Tamil Websites and WebPages
— Work to be done/Time Frame
— Next Symposium - TamilNet'98

Tamilnet 1997 Conference
List of paper presentations

International Symposium for Tamil Information Processing and Resources on the Internet, 17-18 May 1997, National University of Singapore, Singapore

Program Schedule
Day 1 - 17 May, Saturday

time event
09.00 Registration/Reception

10.00 **Conference Opening & Welcome Addresses**
-- Host, Dr Tan Tin Wee, Internet R&D Unit, National University of Singapore
-- Co-Host, A/Prof Koh Tai Ann, National Institute of Education
-- Opening Address by Conference Chair, Mr Naa Govindasamy, National Institute of Education
-- Speech by Guest-of-Honour, Prof S Jayakumar, Minister for Law and Foreign Affairs
10.30 Break for Refreshments

11.00 **Session 1 - Tamil encoding schemes and keyboard input methods**
-- An overview of Different Tools for Word-processing of Tamil and A Proposal Towards Standardization

by Kuppuswamy Kalyanasundaram, Swiss Federal Institute of Technology, Switzerland
-- ISCII and Tamil - A Perspective
by N. Anbarasan, Applesoft, Bangalore, India
-- Sri Lankan Experience of Development of Tamil Input/Display Methods
by S.T. Nandasara, Institute of Computer Technology, University of Colombo, Sri Lanka
13.00 Lunch

14.00 **Keynote Speech**
-- The Development of a Tamil Internet: Possibilities and Challenges for the Future
by Harold Schiffman, Penn Language Center, Univ. of Pennsylvania, USA

14.30 **Session 2 - Towards a standardization…**
-- Tamil and Computer
by Sujatha, TamilNadu standardization committee, India
-- Selection and Standardization of Tamil Keyboard Layouts: Recommendations of Tamil Nadu Standardization Committee
by S Kuppuswami & V Prasanna Venkatesan, Computer Studies, Pondicherry University, India

15.30 Break for Refreshments

16.00 Singapore's TamilWeb presentation
by Naa Govindasamy, National Institute of Education, Singapore

16.30 **Session 3 - The TamilNadu Experience**
-- Towards Tamil Xanadu - First Step: An hyper news service in Tamil
by S. Senthil Nathan, Madras, India
-- My Experiences with Electronic Publishing
by Maalan, Kumudam, India
17.30 End of Day 1

Day 2 - 18 May, Sunday
Time Event
10.00 **Special Address by Thamizh Kudimagan**,
Hon'ble Minister for Tamil Language and Tamil Culture,
TamilNadu Government, India

10.30 Presentation of the Tamil Nadu Budget 1997 Live via the Internet
by Mohammed Yunus, Cyber Globe, India
11.00 Break for Refreshments

11.30 **Session 4 - Research on Tamil text archival, search & retrieval**
-- iAgent: A System for Managing Networked Tamil and Multilingual Information Resources
by K. Rajaraman & Lai Kok Fung, Information Technology Institute, Singapore
-- Significance of Creation and Use of Corpus of Modern Tamil Prose Text through the WEB by Vasu Renganathan, University of Pennsylvania, USA

12.30 Lunch

14.00 **Session 5 - Tamil and Unicode and Implementation**
-- Unicode and Tamil: Issues with Implementation
by Muthelilan Murasu Nedumaran,
SunSoft, Sun Microsystems (Asia South region)
-- The Brand New World of Tamil Home Page: Creating and Browsing Tamil Homepage
by Chong Chiah Jen, Star+Globe Technologies, Singapore

15.00 Break for Refreshments
15.30 **Session 6 - Tamil Educational Resources**
-- Incorporating Internet Tamil teaching resource website into the Tamil language curriculum
by Balasundaram Mahadevi, St Hilda's Primary School, Singapore
16.00 Discussion Forum
17.30 End of symposium

Speech by Prof. S Jayakumar, Minister for Law and Foreign Affairs, Guest of Honour at the Opening Ceremony of TamilNet'97

Jointly organised by the National University of Singapore and the National Institute of Education of Nanyang Technological University on 17 May 1997 at the Auditorium, Computer Centre, NUS.

The Honourable Minister for Tamil Language and Culture of Tamil Nadu, Dr Thamiz Kudimagan, ladies and gentlemen. I am pleased to be with you at this first International Symposium on Tamil Information processing and resources on the Internet. I am heartened to see so many participants and speakers from all over the world gathered in Singapore to discuss and exchange ideas on how to promote the Tamil language.

We live in an information age. Efficient access to, and use of information, will be a key determinant of competitiveness. The Internet is a key device today for that purpose. The Singapore Government has recognized its importance. Hence the heavy emphasis we place IT in our schools. The dominant language of the Internet is English. It is essential therefore that as a nation and community we maintain our principal proficiency in the language.

But Internet also offers an environment within cyberspace for other languages to thrive. It creates an international electronic community of the users of the language (in this case Tamil) and gives access to global resources. This is particularly important for small minority communities (**like the Tamil-speaking group in Singapore which forms 65% of the Indian population**) to have such global access. It helps keep the language and culture vibrant. TamilNet is therefore a welcome step in this direction.

The partnership of technology and language is not new. Indian language printing in India was first introduced by western Christian missionaries in the 16th century through the printing of a church prayer book in the Tamil language. In time, through this new technology, the existing palm-leaf written literature and the wealth of other literary materials soon found their way into more permanent form of printed books.

Today, through Information Technology and global Internet, the Tamil language now finds a place in cyberspace. Again, although the technology was introduced by the West, I am delighted to learn that homegrown Singapore technology has played a catalytic role in this.

I understand that a prototype of Tamil Internet technology was tested when an electronic selection of representative poems from our official languages from the anthology Journey: Words, Home and Nation - Anthology of Singapore Poetry (1984 -

1995) was launched by the President of Singapore, HE Mr. Ong Teng Cheong in October 1995.
By February 1996, we were able to launch a Tamil Internet system called TamilWeb and since then, Tamil language teachers, researchers and users both nationally and internationally have been able to email each other and surf the internet in Tamil. Introduction of the tools for the total Internet Tamil solution (creating, browsing and searching web pages, and emailing in Tamil and English) by the researchers from the Internet Research and Development Unit of National University of Singapore jointly with the National Institute of Education, Nanyang Technological University, paves the way for a lot of possibilities on Web publication. The Tamil publishing industry will also never be the same again.

When the Internet Research and Development Unit of NUS hosted SindaLink, the Singapore Indian Development Association's newsletter on the Web in June 1996, it was among the first newspaper to go on the web in the text mode, inclusive of graphic picture display. The Educational databases and the Interactive Web pages introduced by the TamilWeb and the University of Pennsylvania for the Learning and Teaching of the Tamil language, have further given new dimension to the learning and teaching of Tamil language. I am happy to note that a few schools are already using these Websites in their classroom teaching. A presentation by a school teacher at this symposium will illustrate the bold move in embracing the new technology for more effective classroom teaching. I am happy to note that these breakthroughs were made possible were made possible by a multi-racial project team from the two Universities in Singapore.

With the TamilNet, the Tamil speaking community in Singapore can pride itself in moving in tandem with technological advances. The TamilNet can help to make the Tamil language a vibrant and relevant language. The TamilNet would reach out to young Tamil Singaporeans. With appropriate content in TamilNet, it can become another important channel through which values and relevant customs and traditions are passed on to the younger generation. The multi-media facilities that Internet and computer technology provides, learning the Tamil language can be further facilitated. Singaporeans and other Tamils around the World can learn and improve their command of the language at their own time and in the convenience of their homes.

Today, more than 20 delegates from six countries and nearly a hundred teachers of Tamil in our schools have gathered here to make future plans, and to discuss and exchange ideas on the new directions and visions rendered possible by this partly

homegrown technology. I welcome all our visitors from abroad and I hope you enjoy your stay in Singapore. I have great pleasure now in declaring open this Symposium.

AN OVERVIEW OF DIFFERENT TOOLS FOR WORD-PROCESSING OF TAMIL AND A PROPOSAL TOWARDS STANDARDISATION

Dr.K. Kalyanasundaram,
Institute of Physical Chemistry, Swiss Federal Inst. of Technology,
CH-1015 Lausanne, Switzerland

(Invited Paper to be presented at the "International Symposium for Tamil Information Processing and Resources on the Internet, National Univ. of Singapore, Singapore, 17-18,May 1997)

Introduction

Dravidian Languages such as Tamil use non-roman letters as alphabets. Typing of text materials in computers of these Indic languages requires use of either specific font-faces and/or word-processing softwares. In this paper, features of some of the most commonly used tamil font faces and softwares are reviewed and a possible scheme towards standardisation of Tamil Computing is also indicated. The term 'Tamil Computing' is used in a narrow sense to cover the area of word-processing of tamil-related materials on computers. Tamil Computing cover a much broader domain with applications in many areas: tools for larger databases of different kinds using tamil script, multi-media kits involving tamil, multi-lingual dictionaries and translation softwares etc.

In the last two decades, many different fontfaces and desk-top publishing (DTP) softwares have appeared for word-processing of Tamil and along with them different typing (input) methods. Some of these are based on simple recasting of the tamil typewriter keyboard in the form of 7-bit fonts. Others are sophisticated 8-bit font/word-processing packages where the actual keystrokes and their relative sequence are interpreted to provide the required tamil texts. These packages allow different modes of input including romanized/transiliterated input. Font Encoding, i.e., the exact location of different tamil characters in the standard extended ASCII table (128 or 256 slots) in the tamil font being used determines the 'output' content of the tamil text irrespective of the mode of 'input'. Tamil text files created using one font/DTP pacakge

cannot be read using another font unless the font encoding scheme is identical between the two fonts in question.

There is a growing number of tamil pages being put on the Internet/WWW using fonts, packages with different font encoding schemes. So we are now in an unpleasant situation: One needs to acquire and install as many fonts as the number of tamil web pages and archives available on the internet. Necessity for setting standards arises also from the growing trend to exchange/share information between individuals placed in different parts of the world. In the absence of any standard protocols by which the information storage is carried out at the font-encoding level, information exchange on a world-wide become too complex for many of the concerned individuals, if not impossible. Majority of the end-users (Tamil community at large) are not well-versed in technical aspects of data storage, transfer. So procedures have to be designed so that ordinary/common people can put up web pages and share information electronically in tamil world-wide without getting involved too much into the technical nitty-grittys. Any proposals for standardisation needs to accommodate the current typing habits/preferences (some kind of backward compatibility).

Transliterated/Romanized form of Tamil

By transliterated/romanized tamil text, we refer to reproducing in a near-close phonetic form, the tamil texts using roman alphabets. Thus, the tamil word for father is written as appA (or appaa), mother as 'ammA' (or as ammaa). Transliterated form of reproducing dravidian language materials has been popular amongst western indologists for well over a century (pre-modern computer Era). Even standards were discussed and adopted in an international conference as early as 1888.

The earliest and widely used transliteration scheme is what is known as Library of Congress Scheme which uses roman alphabets with diacritics (horizontal bars or circles added above or below roman alphabets) to represent alphabets of dravidian languages. Figure 1 shows pictorially this and other transliteration schemes for Tamil discussed in this paper.

Fig.1 LC and plain ASCII type Transliteration schemes for Tamil

Diacritical markers added to a letter or symbol show its pronunciation, accent, etc., typically indicating that a phonetic value is different from the unmarked state. The scheme is very general in scope and hence can be used in all of the indic languages. Established tamil research centers all around the world are aware of this scheme and

most of them implement this scheme as such without modifications. In Chennai, Institute of Asian Studies (engaged in publishing many of the tamil literature related research) and Roja Muthaiah Tamil Research Library with links to Univ. of Chicago (involved in electronic cataloguing of 50000+ precious Tamil books collections) are examples of institutions that follow this scheme.

Transliteration schemes for writing romanized Tamil

Vowels

அ	ஆ	இ	ஈ	உ	ஊ	எ	ஏ	
a	ā	i	ī	u	ū	e	ē	(LC)
a	A	i	I	u	U	e	E	(Madurai)
a	A	i	I	u	U	e	E	(Kōln)
a	aa/A	i	ee/ii/I	u	oo/uu/U	e	E/ae	(Adami)

ஐ	ஒ	ஓ	ஔ	ஃ	
ai	o	ō	au	ḵ	(LC)
ai	o	O	au	h/q	(Madurai)
ai	o	O	au	H	(Kōln)
ai	o	O	au/ou	q	(Adami)

Consonants

க	ங	ச	ஞ	ட	ண	த	ந	
k	ṅ	c	ñ	ṭ	ṇ	t	n	(LC)
k	ng	c	n~	T	N	t	n'	(Madurai)
k	g	c	n^/jn	T	N	t	n	(Kōln)
k/g	nG/ng	c/ch	NY/ny	t/d	N/Nn	th/dh	n^	(Adami)

ப	ம	ய	ர	ல	வ	ழ	ள	ற	ன	
p	m	y	r	l	v	ḷ	ḻ	ṟ	ṉ	(LC)
p	m	y	r	l	v	z	L	R	n_/n2	(Madurai)
p	m	y	r	l	v	z	L	R	n_/n2	(Kōln)
p	m	y	r	l	v	z/zh	L	R	n	(Adami)

Given the practical constraints on the scope of present day electronic communications (largely 7-bit) alternate transliteration schemes based on plain ASCII characters have also been in use widely. Figure 1 also includes some of the commonly used transliteration schemes of this kind. Plain ASCII scheme was considered in the early pre-computer era but was abandoned as being non-practical. In the last two decades with the growing use of computers, there is an increasing number of individuals and institutions that employ some form of a 'transliteration scheme' based on plain ASCII

roman characters. Presently most of the postings on the USENET Newsgroups of Internet such as soc.culture.tamil quote tamil texts in the form of romanized text, for display on plain ASCII terminals. MADURAI software uses a code to construct tamil alphabets on screen in four lines using ASCII letters. Though it is not "print quality" it allows to convey the message in quasi-tamil script. The classic 10-volume reference work "Tamil Lexicon" published by the Univ. of Madras during 1929-1939 used the transliteration scheme based on plain ASCII. The Institute of Indology and Tamil Studies of Univ. of Cologne (K?ln) uses this shceme for the cataloguing of their 50000+ tamil books collections and also for their extensive collection of electronic texts of ancient tamil classics (e.g, Sangam Literature).

As said earlier, writing in the LC form of transliterated tamil on Computers requires special fonts that contain roman letters with the diacritrics. Library of Congress and major Tamil libraries in the USA and Europe allow on-line search of their catalogues from anywhere in the world. In order that searches can be made using simple ('dumb') terminals, on-line catalogues allow search using plain ASCII characters without the corresponding diacritical markers. Thus, one has to use keyword 'anil' for squirrel while searching LC or Univ. of California, Berkeley. But, at the IITS library of Univ. of K?ln where the indexing is on alternate transliteration scheme (based on plain ASCII), the search would be as 'aNiL' ! Thus, here we have an anamolous situation where care has been taken to catalogue books using a special font (not readily available) but all its features are lost while doing search using plain ASCII characters. There is also the practical problem that one has to first educate oneself as to which form of transliteration scheme used at the place of search.

In view of the above points, it is essential that, some consensus be reached on a universally adopted transliteration scheme. As will be discussed below, there are now DTP softwares that allow 'input' in romanized text format. Here also it would be better if some standard form of transliteration scheme is universally adopted. Our preferences are for a scheme such as that used in Adhawin/Madurai, one that allow writing in near phonetically equivalent form but using plain ASCII characters.

Word Processing using 7-bit tamil fonts (direct output)
Since tamil typewriters have been in use for many years before the advent of computers, it is logical that early approaches to tamil computing involved implementing the classical typewriter in the form of 7-bit fonts. Various tamil characters are placed under different roman letters at the equivalent locations of the tamil typewriter. All of the tamil alphabets are obtained by using the normal and shift-mode operation of the standard keyboard. While some of the alphabets are obtained in single keystroke,

others are obtained by two or three keystroke operations. With such tamil fonts, those who are accustomed to typing on tamil typewriter can make the transition to tamil computing without difficulty and loss of any typing speed. This trend is very strong in Tamilnadu even today. Majority of tamil computing use the tamil typewriter keyboard layout(s). So any Tamil Computing Standardisation efforts need to take this reality into account. There are many fontfaces of this type available: TAMILLASER of Prof.George Hart, ANANKU of P. Kuppuswamy (widely used in continental US), SARASWATHI of Vijayakumar (widely used in canada) are some examples. BHARATHI word processor for plain DOS computers was one of the early ones to appear (in early eighties) in Malaysia and Singapore region. VENUS is a recent, updated version of this word-processor running under Windows environment.

The common logic in any keyboard layout design is to have most commonly occurring letters placed in the central/middle part of the keyboard (and less frequent ones moved to left/right extremes). This concept/logic was applied quite a while ago in the design of typewriters. In Tamil, in good old classical tamil typewriter layout, one particular assignment was chosen:
middle line ya, La, na, ka, pa, modifier for aa, tha, ma, ta in middle line; nga, Ra, n^a, ca, va, Na, ra, sa, zha, modifier for i in the top line and ii, la, o, u, e, ti, modifier for e, a, i at the bottom line. There have been many re-examination of this concept of character placing for tamil keyboard recently. Mohan Tambe of CDAC, Pune designed a keyboard layout for indic languages using such an analysis. Naa. Govindasamy (host of this conference) has made similar analysis for tamil and has designed the Kanian/IE/Singapore Tamil Keyboard layout.

An alternative approach to tamil typewriter keyboard layout involves phonetically linking tamil characters to be typed to corresponding roman letters. Thus you hit the key k to get ka, m for ma, l for la, p for pa, k followed by i for ki, k followed by I to get kii and so on. For those who never used the tamil typewriter, this approach can be intuitive and very appealing. Since tamil characters of 7-bits are readily accessible via normal and shift-modes of the keyboard on all computers, I designed a phonetically based 7-bit font called MYLAI. The term 'phonetic' is used in a slightly different context by many (e.g participants of this conference Naa. Govindasaamy, Ravindran Paul). So we would use the abbreviation WYTIWYG (what you type is what you get) layout to refer to keyboard layouts based on the above cited phonetic input method. The frequency of occurrence of tamil characters in tamil need not necesarily be the same as in English. So I had some reservations on sustained interest for people to use keyboard layouts of the WYTIWYG kind. To my pleasant surprise, the reception to Mylai keyboard has been overwhelming. In the last three years, several thousand tamil lovers all around the world

have received a copy of the Mylai font and happily using it for tamil computing. Some even wrote to say that, with the satisfaction in Mylai, they have been deleted some tamil font faces of classical typewriter kind that they bought earlier for a price. I should state here that, mylai was not the first tamil font available free on the internet (there have been several others freely available) nor it gives the most aesthetically pleasing print out for very demanding end-users.

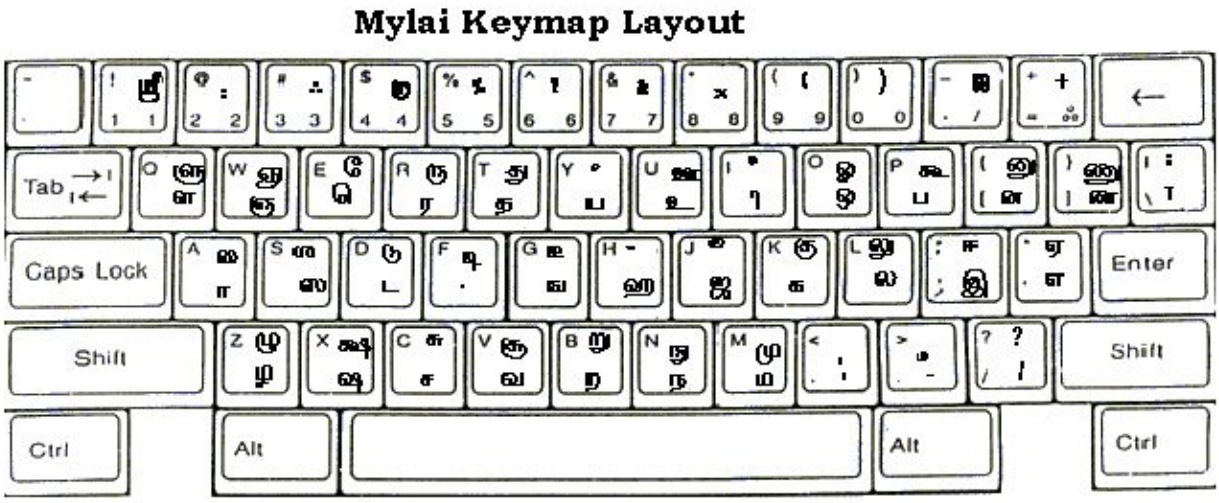


Fig.2 Mylai phonetic/wytiwyg keyboard Layout

Word-processing using 8-bit font faces (direct output)

If one counts the number of alphabets of tamil, we have over 230 (13 vowels, 18 consonants and products (uyirmeis) derived from these. Tamil is one of the Indian languages where many of the compound (uyirmei) alphabets have complex geometric structure (glyph) of their own. In 7-bit fonts with 128 slots, nearly half of them are not available for placing tamil characters (first 32 slots reserved for control characters, 10 places for roman numeral and another 10 or 12 for various key punctuation marks). For the number of tamil alphabets to handle, the remaining positions are rather limited. In 7-bit fonts, a number of compound/uyirmei letters are obtained simply by adding a modifier glyph to the parent consonant. Tamil typewriter uses this concept extensively. 'Kerning' is a technique that allows controlled fusion of two successive character. Unfortunately, kerning is not easily implemented on many computer platforms. Without kerning, the quality of the output for on-screen display and in print using such 7-bit fonts can be far from satisfactory, at least for commercial publishing houses. So, there have been efforts to go for fonts of the 8-bit type (256 slots available). 7-bit and 8-bit fonts have their own merits and demerits. We will return to this topic later on.

In the absence of kerning and other character control features, in many of the software packages designed for publishing houses, many of the tamil uyirmeis with complex structural forms are included as such in the upper ASCII part (128-255). This way aesthetic quality print can be ensured. In the Macintosh OS, it is easy to access many of these characters in the upper ASCII part using the 'option' and 'shift-option' keys. T. Govindaraj (of USA) designed a 8-bit tamil font for Mac called PALLADAM making use of this feature. In this font design, tamil alphabets ma, mu and muu, for example, are obtained using the keys m, shift-m and option-m respectively. In Windows, one needs to have the 'alt' key down and type in the three digit reference number of the character in question preceded by a zero, as in 0172 or 0213. One needs to remember these numbers to be able to type at reasonable speed. So keyboard editors/managers are often used. With these keyboard editors, one can access any character using any key irrespective of the font encoding scheme used.

RAMINGTON TAMIL is an example of the 8-bit extension of the classical tamil typewriter keyboard. In addition to 26 slots occupied by roman numerals (10) punctuation marks (11) and mathematical operators (5), 78 tamil characters are placed in the font face. On Windows-based PCs, the alt-key is used to obtain those extra tamil characters. Softview Computers of Chennai markets tamil word-processors that work on the Ramington Tamil keyboard. Fontfaces with this Ramington Tamil keyboard layout are used extensively by the publishers of Tamil Newspapers and Magazines of Chennai.

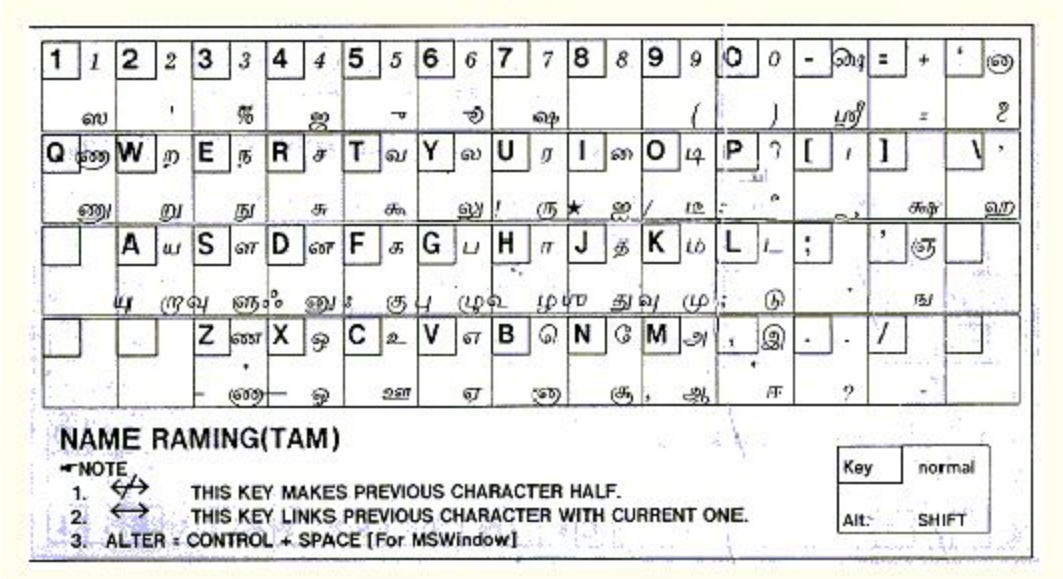


Fig.3 Ramington Tamil Keyboard Layout

Word Processors based on romanized input (intepreted output)

ADAMI was one of the early tamil word-processors for MS-DOS PCs produced by Dr.K. Srinivasan of Canada in early eighties (released in 1984 for CPM-80 computers) to recast such transliterated text into Tamil. The tamil text is to be typed using a plain ASCII transliteration scheme. Upon compiling/execution of the linked macro, this romanized text page is recast on screen in equivalent Tamil. One needs to return to the romanized text mode to make the corrections if any. In a more recent version of this software called THIRU, the author provides a split screen, where the roman text being typed in the bottom half of the screen is continuously recast in the upper half in Tamil. ADHAWIN is another recent implementation of the same software but for Windows-based PCs. The transliteration scheme used in MADURAI is a subset of that used in ADAMI/ADHAWIN. The software operation used here is part of a general classification scheme called "romanized input/interpreted output" package.
For those who never wrote extensively in Tamil (and beginners who are not sure of exact uyirmei to use in writing tamil worlds, e.g. na/Na), word processors that allow transliterated input is attractive. Adami, Madurai, ITrans, XLibTamil softwares mentioned earlier to this category. The last three freewares are popular amongst the UNIX user community. They are being used widely to make tamil-related postings in USENET newsgroups. Used in conjuncture with corresponding meta-fonts and TeX-type word-processing extensions, high quality print-outs can be obtained for the tamil texts.
MURASU, ANJAL word-processing packages widely used in Malaysian, Singaporean Tamil Newspapers and Magazines are the products of Muthu Nedumaran present at this conference. These packages belong to the group of "romanized input/interpreted output" tools. The inaimathi and related fontfaces used in these pacakges are of the 8-bit bilingual type. The first 128 (0-127) slots are filled by roman characters as in basic ASCII and the tamil characters occupy the upper ASCII slots (128-255). By invoking the keyboard editor it is possible to access either of these two blocks. In the tamil typing mode, the roman keyboard strokes and their relative sequence are continuously interpreted to present equivalent Tamil characters on screen. Thus you type 'kathai' to get the equivalent tamil word.

Word Processors based on phonetic keyboard input (interpreted output)

There are now available intelligent tamil word-processors where the large number of uyirmei alphabets are obtained by a sequencial keying of the corresponding mei and uyir characters. Thus the keystrokes for consonant k followed by vowel i leads to

appearance of compound character ki. Keyboard layouts of this kind have been called "phonetic". There are no characters for kokki's kombu's etc. The keyboard driver does the mapping and remapping based on the sequence of keypress events. An advantages of this approach is that the number of keys to use to get all the uyirmeis are considerably less. Mohan Tambe (formerly of the Centre for Developments for Advanced Computing CDAC) was one of the early pioneers working on the keyboard layouts appropriate for indian languages. His phonetic keyboard layout known as INSCRIPT was initially designed (in 1983) for Devanagiri script input. This has been adopted for use in the multi-lingual word-processors CDAC developed for indian languages (cf. references to CDAC and Inscript in the next section). THUNAIVAN word-processsor of Ravindran Paul, IE PHONETIC KEYBOARD LAYOUT of Naa. Govindaswamy, CHARACTER PHONETIC DEPENDENCIES/YARZAN keyboard editor of R. Shanmugalingam are different forms of implementation of this phonetically based keyboard input concept. When compared to wytiwyg keyboard layouts, phonetic layouts reduce considerably the number of keystrokes required to get the o-kara, oo-kara, ou-kara uyirmeis.. Here you will type k followed by o or O. Two keystrokes give 3 characters.

Place of Tamil in Multilingual word processing packages

Thanks to advances in the design of faster memory chips and compact high capacity storage devices, computers with Gbs of storage (hard disc), Mbs of RAM memory and high speed (>200 MHz) are already available at very affordable prices for the general public (at leat in the western world, if not in India). To make full use of this capability, multilingual packages are being developed that allow preparation of documents containing scripts of more than two or three languages. Several thousand characters corresponding to ten or more languages are bundled up in a single 'super-font' and appropriate software allows selection of one of these languages from a pull-down menu. The viability of this approach has already been demonstrated in the multi-linguage kit covering all the European languages, Greek and Turkish for Windows 95/NT environment. Microsoft currently distributes 'free' a font face containing 800+ characters and also a software to use along with it. Indian languages are not yet included in this multilanguage kit of Microsoft. MtScript (developed by Univ. of Aix-en-Province under support of French CNRS) is a multi-lingual text editor (for UNIX running Solaris) that enables using several different writing systems (Latin, Arabic, Cyrillic, Greek, Hebrew, Chinese, Japanese, Korean, etc.) in the same document. All of the languages defined in the ISO 8859-X schemes are supported in this package.

Unicode Consortium is currently working on a world-language standard character set (ISO 10646) for future use for multi-lingual wordprocessing. Unicode 2.0 version currently under discussion proposes specific slots character assignments for world languages including all indic languages (devanagiri, gurmukhi, tamil, telugu, malayalam, kannada,....). Muthu Nedumaran's paper at this conference dwells in to the details of implementation of this Unicode package. So we will not go into its detail except make a few remarks on the implications of the proposed character set (font encoding scheme). It was mentioned earlier, that, 8-bit fonts allow a large tamil alphabets (100 or more) stored in their native form and this in turn allows high quality production of printed tamil texts required for commercial publications. Unicode character set for Tamil has the bare minimum (64)- vowels, consonants, tamil numerals and a handful of modifiers to add to the consonants to get the compound (uyirmei) characters. None of the uyirmeis have been allocated any slot. If one uses only the above minimal character set, many of the uyirmeis have to be written in new forms (e.g, write pu, mu, puu, muu using the same right modifiers that are added to grantha letters ha/sa to get hu/su or huu/suu). Writing many of the uyirmeis in this new form (and deleting all the currently used structural forms/glyphs) in essence, amounts to introducing drastic language reforms - reforms in the way the script of the language is written currently.

In a parallel development to Unicode, the Dept. of Electronics of the Govt. of India has been developing standards for computing in Indian Languages (including Tamil) for over a decade. The primary tool is Graphic and Intelligence based Script Technology (GIST), a phonetic based computing technology. Center for Development for Advanced Computing (CDAC) based in Pune is the organization in India engaged in developing multi-lingual computing tools based on the GIST technology. Mohan Tambe (working intially at IIT, Kanpur, later as the Head of the GIST group at CDAC, Pune) is the brain behind the major multi-lingual computing projects for indian languages in India. The 1986 proposals of DOE for possible font encoding standards were revised by the Govt. of India in 1988 and were adopted as the 'national standard' under the name "Indian Standard Code for Information Interchange (ISCII-88). The early version of the Unicode apparently was modelled on the ISCII-88 standard. As in the Unicode scheme, the basic characters defined in the ISC character set is graphics characters as (in Hindi) Anuswar, Visarg, a set of vowels, set of consonants and vowel signs. The display rending and formation of conjuncts is left to the softwares meant for such purpose. Along with the ISCII standard for font encoding, the "phonetic keyboard layout" of Mohan Tambe has been adopted under the name INSCRIPT as the national standard for keyboard layout. The GIST technology works in the 8-bit mode where the tamil (or any indian language) characters are placed in the upper ASCII slots 160-255 (actually 79 characters/glyphs). The entire lower half and the line drawing character set in the

upper half are left undisturbed for English so that bilingual documents consisting of English and the indian language can be readily prepared. CDAC markets several products for multi-lingual computing based on this GIST technology. The phonetic/inscript keyboard designed by Mohan Tambe is used in all of the CDAC/GIST packages. Apex Language Processor (ALP), ISM (ISFOC Script Manager), LEAP (Language Environment for Aesthetic Publishing) are some of the multi-lingual word processors sold by CDAC directly or through its franchises. Popular word-processing package SHREE LIPI of Modular Systems is another commercial version of the package. LEAP is a multiscript word processing package for windows (like MS-WORD) that allows comparing texts in all indian languages. This is a cost effective solution for marketing/advertising agencies where trade literature giving details of the products can be given in all indian languages one after the other.

Apple has released very recently for Macintosh computers, a premier version of its 'INDIAN LANGUAGE KIT (ILK)' . This package contains fonts/software for word-processing in Devanagiri and Gurmukhi. It has been stated that the ILK package is modelled on the ISCII standards of the Govt. of India. INSCRIPT is the generic name given for the keyboard layout specifically designed for input of indic languages. ComStar of Cupertino, California, USA markets multi-lingual Word Processors called Gamma UniType and UNIVERSAL WORD FOR WINDOWS that allows preparation of a multi-lingual text and the package supports a large number of world languages including tamil. WordMate (also of ComStar, Inc) is a multi-lingual versatile software /keyboard driver that enables the user to tupe any of a long list of languages directly into virtually any windows application.

The Multilingual directory of the Internet lists the following softwares currently available for multi-lingual word-processing including tamil: Allwrite (of ILECC), Chitralekha (of Modular systems), Apex Language Processor (ALP) and ISM (ISFOC Script Manager of CDAC, Pune), Amicus (of Amicus), Gamma UniType and Multilingual Scholar (of Gamma Productions), Kalam (of Solustan Inc), LEAP (Language Environment for Aesthetic Publishing, of CDAC, Pune), Prakashak (of Sonata), Swadesh(of Institute for Typographical Research), Vision Publisher (of Vision Labs).

Proposals for standardisation/font encoding for Tamil should taken into account the mode of functioning of these multi-lingual word-processors. It would be unwise and non-practical to have different world standards for tamil - one for mono/bilingual usage within the 8859-X scheme and one for multilingual packages.

In part I, various tools that are available today for tamil word-processing were reviewed. Herein we attempt to classify them in some unifying framework and use that framework as the basis for proposing a standardisation scheme for Tamil word processing.

Classification of tools for Tamil Word Processing
Functioning of any word-processor can be divided into two parts - those connected with the 'input' process and those with the 'output'. Based on the features of the input, output processes involved it is possible to classify all word-processing tools into following categories:
classical typewriter input/Direct output ;
wytiwyg ("what you type is what you get") input/direct output;
romanized input/interpreted output ;
phonetic input/Interpreted output.

Table 1 lists examples of different font faces and word-processing softwares grouped according to the above classification.

Name	Author	Type	Platform,Remarks
Fontfaces			
[ananku]	P. Kuppuswamy	direct/ttw classical-1	win/mac,7-bit
tamillasr	George Hart	direct/ttw classical-1	mac, 7-bit
[saraswathi]	Vijayakumar	direct/ttw classical-1	win, 7-bit
[TMNews]	(dinamani)	direct/ttw classical-2	win, 7-bit
[amudam]	(softview comp.)	direct/ttw classical-3	win, Ram.ttw
mylai	K. Kalyanasundaram	direct/wytiwyg1	win/mac/unix, 7-bit
mylai-sri	K.Srinivasan	direct/wytiwyg1	win,mac, 7-bit/
palladam	T. Govindaraj	direct/wytiwyg2	win,mac, 8-bit
valai-sri	K. Srinivasan	direct/wytiwyg3	wind, 7-bit
trutamil	Raja Seshadri	direct/wytiwyg4	win, 8-bit?
Word Processors			
anjal/murasu	Muthu Nedumaran	interpreted/romanized1	win, unix, 8-bit
adhawin/Adami	K. Srinivasan	interpreted/romanized2	win, 8-bit
[nalinam]	Sivaguru Chinniah	interpreted/romanized3	win, 8-bit
ITrans	Avinash Chopde	interpreted/romanized4	Unix/win, 8-bit
XLibTamil	G. Swaminathan	intepreted/romanized4	Unix, ?
madurai	Bala Swaminathan	interpreted/romanized2	unix/PC, ?
PCTamil	Vasu Ranganathan	interpreted/romanized3	DOS PC, ?
[IE/phonetic	Naa. Govindasamy	interpreted/phonetic1	win/mac/unix, 8-bit
[Thunaivan]	Ravindran Paul	interpreted/phonetic2 win, 8-bit	
[Yarzan]	Shanmugalingam	interpreted/phonetic3	win, 8-bit
[LEAP]	CDAC	interpreted/phonetic4	win, 8-bit
[Gamma UniType	ComStar	interpreted/?	win
bharathi	?	interpreted/?	DOS PC
venus	?	interpreted/?	win

Examples of 'Direct' tools are font faces used with associated keyboard layouts in typewriter or WYTIWYG format. In direct usage of simple font faces, the 'output' has a one-to-one correspondance with the input. For every keystroke, there is a character output. There is no software interpretation or intervention of the keystrokes. What letter you see on screen depends on what letter is stored under the keystroke in question. In other elegant word-processors, the 'input' is 'interpreted' by the software to give the output. The input can be in the form of romanized text or phonetically based. In all cases, keyboard editors/managers allow some manipulation of the input process. 'Font-encoding' determines the final 'output'.

A primary requirement for any standardisation process is to have a standard font encoding scheme. Irrespective of the mode of input and the output modes, all word-processing tools must use this unique standard character set. With such a unique font encoding standard, it is enough to have one single tamil font/word processor to exchange tamil documents electronically (including via WWW pages of Internet). In any implementation of standards, there is geneuine fear on the part of end-users to know how much of their current typing habits and word-processor capabilities are to be sacrificed. Is it possible to find methodologies by which different typing habits ('input' practices) of end-users and also the choice of different modes of inputs be guaranteed within such a scheme imposing a standard character set (font encoding scheme)? The answer is yes. The details are elaborated in the following paragraphs.

Keyboard Layouts and associated Editors/Managers

Keyboard Editors (or Managers) allow access of any character stored in the font-face by typing any key on the keyboard. Different keyboard layout can handle typing preferences of individuals. As an end-user/laymen, we do not care where ku is assigned in the ref. table. But we would like to know which key on the keyboard we should use to get ku, ki and so on. Keyboard layout is what controls this access pathway and hence this is an important point that affects the end-user (you and me!!!) drastically.
Both on Windows and Mac OS, standard keyboard layout softwares come with the system. These allow us to choose different layouts for typing in different european languages. In French, for example, there are at least 4 different keyboard layouts available (french, french-numerical, swiss french, canadian french). Here in the french speaking part, we set my Mac/PCs to use Swiss-French keyboard layout and switch to US keyboard layout whenever typing is done in Tamil using Mylai tamil font. Switching between keyboard layouts is rather trivial (even our 8-year old daughter knows how to change-to go from one keyboard to the other). (In french french keyboard, common french letters like e(accent grave), e(accent acute) etc are in normal numerals key

positions and you need to use shift to get the roman numerals themselves!!). Different keyboard layouts present different characters at key positions of choice. Thus the same fontface set can be accessed differently on the keyboard using different layout schemes.
In tamil, we can have different keyboard editors that allow us to do the same thing. Anjal, for example, can have different keyboard layouts made available to allow tamil typing corresponding to different typing habits. In fact, Muthu Nedumaran is working towards providing a Mylai keymap typing option in future versions of Anjal. In principle, we can have many tamil keyboard layouts (as is the case I mentioned earlier for french) to satisfy every interest.The Summer Institute of Linguistics of Dallas, Texas, USA makes available already in public domain a handful of softwares that allow development of dedicated keyboard editors/drivers for windows (e.g, KeyMan) and Macintosh (SILKEY) platforms. It is desirable though, to limit the number of keyboard layouts to make the life of software designers easier. Too many keyboard options means, the softwares have to be adopted to handle different schemes.

A proposal Towards Standardisation

In the introductory section of this paper it was pointed out that internet is rapidly becoming a main channel/forum for exchange of information worldwide and that currently, tamil on internet is in a rather messy situation. Why? The Indian Government might have proposed a standard character set (ISCII) and a keyboard layout (INSCRIPT) for indian languages nearly a decade ago. However the existence of these standards were not popularised outside India. Due to lack of communications, dedicated softwares for tamil word-processing have been developed independently in India and abroad (particularly in Malaysia/Singapore region where there is a high concentration of tamils). Many of these softwares have their own novel features. Even if the number of tamil-speaking community outside India may be less than those within India, major fraction of the former group have access to computers. Tamil computing is fast catching up in this group. Due to varying font encoding schemes used in the word-processing tools currently employed, the web pages require prior downloading of as many fonts as the web pages to browse. Along with the use of softwares comes the typing habits of individuals. In order to make information exchange of tamil materials via Internet a real pleasure for all of us, it is essential that efforts are taken to unify these different approaches under some umbrella scheme.

It was mentioned earlier that, for a given font, different keyboard layouts can be presented to the end-user through the use of keyboard editors/managers. Given this possibility, one possible approach towards standardisation is to go for a single font

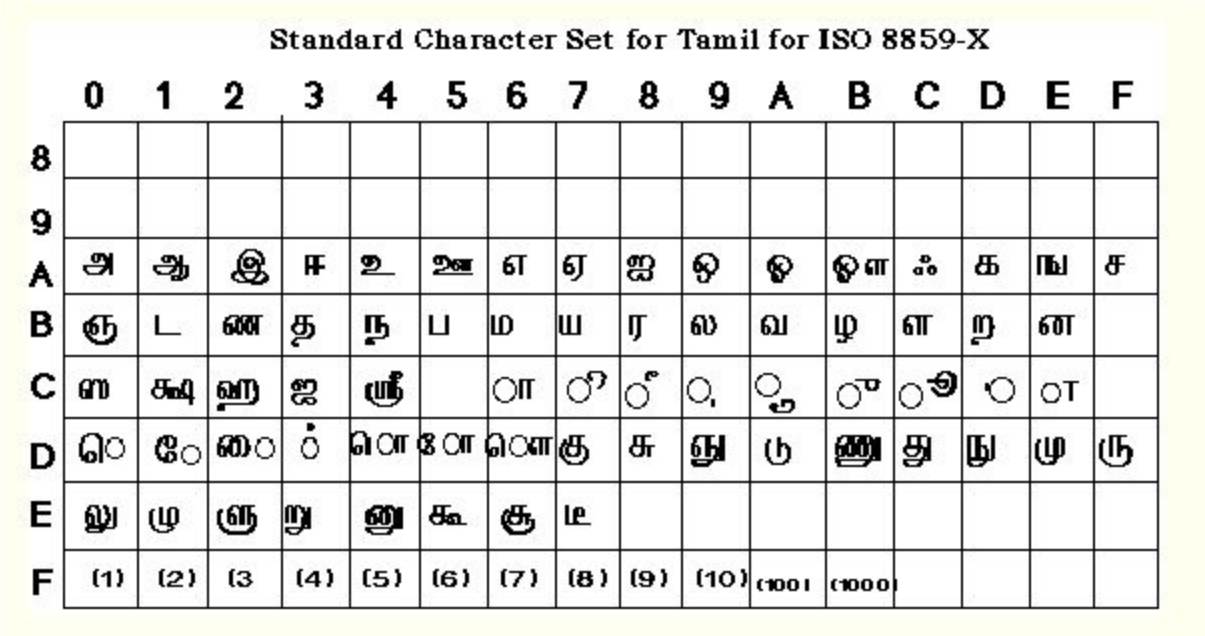
encoding (standard character set) to be adopted in all tamil font faces, word-processors and DTP packages. Each font/DTP package can come with different options of input methods provided in the form of different keyboard layouts made available under a pull-down menu. The feasibility of having keyboard editors/managers for some of the commonly used input methods has already been shown. Thus standardisation process reduces to deciding on a standard character set with some recommendations on possible keyboard layouts that can be provided for users' choice. Since end-users can continue to work with their own favourite keyboard layout, there will not be any major resistance to the implementation of this standard. The major task will be for the software developers to recast their existing font faces to correspond to one standard encoding scheme and provide appropriate keyboard editors/managers that are currently used. With the option available for anyone to test out different keyboard input methods, there can be hope to reduce the number of these layouts (weed out some less popular ones). Implementation/large scale acceptance of proposed standards in a short span of time is possible if and only if the implementation process does not get bogged down with legal constraints imposed by copyright protections and associated high costs to obtain rights of usage of the technology involved in the standardisation process. Clearly, any scheme for font encoding or keyboard layout that is not strictly in public domain can cause problems. Open standards for all the key elements is critical. Recommending possible standards for world-wide tamil computing that are based on propriety materials of few author(s) amounts to patronising and against all free market practices. To facilitate the process and to avoid ambiguity, it is highly desirable that all key players (software developers) in the field openly declare their consent to work within such "open standards framework".

A possible standard character set under 8859-X scheme

Currently most of the world languagues are handled under different standard character sets registered with International Standards Organization (ISO) under different headings. ISO 8859-X of ECMA (European Computer Manufacturers Association) is the most popular of these schemes for handling european and other languages of the world. It is currently implemented by the commonly used web browsers. So, in short, it is a proven technique. The standard (default case for most web browsers) is 8859-1 and this supports most of the languages of Europe and Latin America. 8859-2 (aka as Latin-2) is designed for Eastern European languages, 8859-3 (aka as Latin-3) is designed for South-Eastern Europe, 8859-4 (aka as Latin-4) for Scandinavia (also covered by 8859-1), 8859-5 for Cyrillic/russian, 8859-6 for Arabic; 8859-7 for Greek, 8859-8 for Hebrew, 8859-9 (aka as Latin-5) is same as 8859-1 except for Turkish instead of Icelandic and 8859-10 (aka as Latin-6) for Eskimo/Scandinavian Languages.

For those who would like to know more about these standard character sets, there are a couple of web sites providing additional information: ISO Alphabet Soup ; Info. on ISO-8859 ; Internationalisation.

Herein, we would like to discuss a possible standard character set for tamil for eventual registration under the existing ISO 8859-X scheme. Figure 2 presents one possible standard character set following the general pattern of 8859-X schemes. The character set contains a minimal set of characters or glyphs that one would need to be able to type tamil texts in a form that will be acceptable to majority of the tamil community. It is modelled on the 7-bit tamil font faces of the classical tamil typewriter and 'wytiwyg' keyboard layouts. Tamil texts can be written in all of the possible current writing practices and also in forms corresponding to some recent proposals suggesting reforms in tamil writing practices.



The time of introduction of standards for tamil computing can also be an opportunity to introduce some of the proposed reforms in tamil writing practices. Any reform/revision should be gradual to have a quick world-wide acceptance. Standards proposing drastic reforms will remain on paper and people will continue to write the way they do now. Hence we have to be very careful in deciding the content of the standard character set: number of glyphs Unicode 2.0 and ISCII Standards contain a minimal set of tamil character glyphs (basic vowels, consonants and a handful of modifier glyphs that add to consonants to give the uyirmeis). The actual generation of the uyirmeis is left largely to the softwares. If there are no standards on the actual number of glyphs to be used in tamil word-processing, the output will be software dependent. All the proposed exercise

of standardisation will be useless. As mentioned earlier, most of the uyirmei alphabets of tamil language have their own unique geometric shape/glyphs. Currently most of the 8-bit word-processors designed for professional publishing houses (tamil newspapers, magazines,...) keep these unique uyirmei glyphs within the 256 character set slots. If we do not specifically include these unique glyphs (many are not easily obtained using the kerning techniques), these schemes necessitate writing tamil in a new radically revised form!!! To ensure backward compatibility and ready world-wide acceptance, the proposed scheme includes many of these unique uyirmeis as such. We are fortunate that, two other paper presentations at this conference (of Muthu Nedumaran and Anbarasan) will address specifically the tamil standards as envisaged under the above UNICODE and ISCII standards respectively.

Acknowledgement
Many of the points discussed in this paper were floated and extensively debated in the internet email discussion forum 'tamil.net' during the last couple of months. I would like to thank Mr. Muthu Nedumaran and Mr. Bala Pillai for making this forum available and Muthu in particular for many fruitful dialogues during the past year. I would like to thank all those who participated in these discussions.

Author
Dr. K. Kalyanasundaram is a native of Madras (oops, Chennai), Tamilnadu where he had his early school, university education. He attended Loyola College affiliated to the Univ. of Madras, from where he received his B.Sc, M.Sc degrees in Chemistry. This was followed by doctoral thesis research in the area of photochemistry done at the Radiation Laboratory of the Univ. of Notre Dame in Indiana, USA (received Ph.D in Physical Chemistry in 1976). After spending nearly 27 months in London, UK as a post-doctoral research fellow of the Royal Institution of Great Britain, he moved to his present location of Lausanne, Switzerland in 1979. He is a member of the teaching and research staff of the Chemistry Dept. of the Swiss Federal Inst. of Technology (Ecole Polytechnique Federale, as it is known locally in the french speaking part of switzerland). While surfing the Internet for a couple of years, he came across the huge amount of electronic archives of ancient literary classics in English. When he could not find anything worth talking about in Tamil available on the Internet, he floated an idea of a 'Tamil Electronic Library' in 1994 to the soc.culture.tamil newsgroup. To facilitate electronic text archiving he developed a tamil font called Mylai and has distributed several thousand copies of this font free via internet. With this font as the base, he started building a collection of etexts of tamil literary classics and the tamil electronic library web site. Thus grew his interests to various aspects of Tamil Computing and its Resources on Internet.

Table 1: A classification of various tools available for tamil word-procssing.

Name	Author	type	platform	remarks
Fontfaces				
[ananku]	P. Kuppuswamy	direct/ttw classical-1	windows/mac	7-bit
tamillasr	George Hart	direct/ttw classical-1	mac	7-bit, mac
[saraswathi]	Vijayakumar	direct/ttw classical-1	windows	7-bit
TMNews	(dinamani)	direct/ttw classical-2	windows	7-bit
[Inscrypt]	(softview computers)	direct/ttw classical-3	windows	Ramington ttw?, 8-bit
mylai	K. Kalyanasundaram	direct/wytiwyg1	windows/mac/unix	7-bit
mylai-sri	K.Srinivasan	direct/wytiwyg1	windows,mac	7-bit/
palladam	T. Govindaraj	direct/wytiwyg2	windows,mac	8-bit
valai-sri	K. Srinivasan	direct/wytiwyg3	windows	7-bit
trutamil	Raja Seshadri	direct/wytiwyg4	windows	8-bit?
Word Processors				
anjal/inaimathi font	Muthu Nedumaran	interpreted/romanized1	windows, unix	8-bit, mac
adhawin, Adami	K. Srinivasan	interpreted/romanized2	windows	8-bit
[nalinam]	Sivaguru Chinniah	interpreted/romanized3	windows	8-bit
ITrans	Avinash Chopde	interpreted/romanized4	Unix/windows	8-bit
Washington tamil font				
XLibTamil	G. Swaminathan	intepreted/romanized4	Unix	?
madurai	Bala Swaminathan	interpreted/romanized2	unix/PC	?
PCTamil	Vasu Ranganathan	interpreted/romanized3	DOS PC	?
[IE/tamilfix]	Naa. Govindasamy	interpreted/phonetic1	windows, mac, unix	8-bit
[Thunaivan]	Ravindran Paul	interpreted/phonetic2	windows	8-bit
[Yarzan]	Shanmugalingam	interpreted/phonetic3	windows	8-bit
[Shree Lipi]	Modular/CDAC-DOE	interpreted/phonetic4	windows	8-bit, multilingual
[Gamma UniType multilingual		ComStar	interpreted/?	windows
bharathi	?	interpreted/?	DOS PC	?
venus	?	interpreted/?	windows	?

i) 1,2,3... indicate variations of the input method/keyboard layouts of a given type.
ii) [....] indicate tools are not of "proprietory" nature (not "OPEN").

Unicode and Tamil – Issues with Implementation

Muthelilan Murasu Nedumaran

Abstract

With the advent of the Unicode initiative, Tamil has found a place among other languages of the world for uniform electronic representation. However, the Unicode character encoding for Tamil imposes significant number of issues to both font and software developers who have been working on their own character encoding for their specific needs. The Unicode Standard 2.0 makes clear distinction between characters and glyphs. This is a big diversion from the one-to-one character to glyph mapping Tamil font developers have been adopting all this while. Not every alphabet in the Tamil alphabet set is assigned a character code in Unicode. Neither does it define all the glyphs required to compose all the Tamil alphabets. This suggests that software developers need to work on specific text-processing algorithms to implement Tamil script in their applications. This paper attempts to provide some insight into the difference between characters, glyphs and fonts and looks at the impact that this model will have on two key areas: i) the use of Tamil on readily available shrink-wrapped software ii) development of new applications in Tamil for commercial use and consumer devices.

Introduction

For a decade or so, there have been numerous independent efforts all over the world to bring Tamil into the electronic media. The lack of a universally accepted standard for Tamil text processing has led to the creation of a wide variety of formats and character encodings. However, most of these efforts were centered on fonts and their use with common word processing and desktop publishing applications. The fonts so developed, had sufficient number of glyphs within them to render Tamil text electronically. In some instances, developers would write keyboard drivers to map Tamil characters on the standard computer keyboard. The applications used however, will have no knowledge of the fact that the language used is Tamil. This had many advantages, the main one being the ability to use commercially available shrink-wrapped software for word processing, desktop publishing and even to some extent data processing “as is”.

The simplest possible solution one would think of in overcoming the different character set and input method issue could be to integrate and define a single encoding for Tamil and, though not so critical, standardise on an input method. There are efforts around the

globe to do this and some of them are getting to the stage of bearing fruits. The Webmasters@Tamil.Net mailing list is a clear example of such an initiative.

This approach is important and is required for most of our immediate needs with regards Tamil computing. However, the Internet momentum, coupled with the global shift towards internationalised software development is pushing for the need to have a globally accepted standard for character encoding that encompasses all the languages in the world – past, present and future. Unicode, which is being accepted by most IT organisations, vendors and users alike, makes this possible. Unicode includes all the characters from all major international standards approved and published before December 31, 1990. Tamil made it into Unicode through ISCII (Indian Standard Code for Information Interchange).

Major system software vendors are implementing Unicode today in their native environments. Examples are Windows NT and Java. A data type defined as a character in Java defaults to a Unicode character. This makes software development with Unicode a lot easier as developers need not worry about the risk of disecting a character into two meaningless bytes.

Unicode makes a clear distinction between characters and glyphs. It only encodes characters and leaves it to the text processing application to map the appropriate glyphs for the defined characters and characters formed by combining two or more of the defined characters. Before proceeding further, it may be appropriate to have a good understanding on the difference between characters, glyphs and fonts.

Character, Glyph and Font

Characters are represented by character codes. When a user creates a document and inputs text, the text is represented and stored as characters. Characters are not visible. In other words, a user does not view or print characters. For a character to be viewed or printed (on screen or paper), it must be represented by one or more glyphs. Traditional character codes use a one-to-one mapping of characters and glyphs. In other words every character that is in the character set is represented by a glyph and these are sufficient to render the entire script of that language.

In Unicode, although every character is represented by a glyph, the set of glyphs we need to render the script could be more than the number of characters defined; which is the case for Tamil. This is why we need special algorithms to map sequence of characters into glyphs. For example the sequence of characters and … will result in the

glyph . Both and … are defined as characters and they are sufficient to represent . As such need not be defined as a character.

A font is a collection of glyphs. Modern font technology, such as TrueType Open, also includes the mapping tables along with the glyphs and can contain glyphs for more than one language in a single font.

Characters and glyphs are closely related, with many attributes in common. However, the distinctions between them make it essential that they be managed by separate entities. ISO/IEC 15285 (working draft) states the characterisation of character and glyphs and their relationship as follows :

A character conveys distinctions in meaning or sounds. A character has no intrinsic appearance.A glyph conveys distinctions in form or appearance. A glyph has no intrinsic meaning. One or more characters may be depicted by no, one, or multiple glyph representations (instances of an abstract glyph) in a way that may depend on the context.The relationship between coded characters and glyph identifiers may be one-to-one, one-to-many, many-to-one, or many-to-many.

Character Sets

A Character set is a collection of characters. However, characters from different language systems can be grouped together to form different character sets. This is done primarily because in the past, character sets can only contain a limited number of characters. Character sets are either single byte or double byte. The section below describes both these.

Single byte character sets

A single byte character set employs either a 7-bit encoding or an 8-bit encoding. A character encoding that uses 7-bits encodes only 128 characters. This is the most universal and is what ASCII (American Standard Code for Information Interchange) uses. The 7-bit code space encodes all the printable characters that we see on a typical US-English computer keyboard. They include punctuation marks, numbers, lower and upper case alphabets and mathematical symbols. As these characters plus 32 other control characters take up the entire 7-bit code space, 8-bit encoding was introduced to include characters other than those we see on the keyboard.

With 8-bits representing a character, we can have a total of 256 characters in a set. The first 128 of the 256 is assigned to ASCII to maintain compatibility, the rest of the 128 (commonly called extended set) is where the action happens for non-English languages. This space is not even sufficient to represent all of the languages required by the European Union at once. As such, many character sets were developed and standardised, each of them having the same characters in the fist 128 space and different ones in the extended space. With the exception of CJK (Chinese, Japanese and Korean), the rest of the world uses an 8-bit encoding scheme.

Almost all Tamil encoding efforts by individual font developers used the extended set. An exception is Mylai which used 7-bits and replaced English alphabets.

The limitation this imposes is, it will not be possible for one to have (as an example) English, French and Tamil in the same document that is stored as plain-text (i.e. without font and other typographical information). This is because; both French and Tamil share the same code space (assuming Tamil characters are encoded in the extended space). It will not be possible to differentiate if an 8-bit character is used for French or Tamil.

Plain text is a necessity because it is about the only form of text representation that is universally accepted for information interchange. Besides, plain text is platform independent, i.e. it can be stored, viewed and manipulated in any word processor, computer system or electronic device.

Double Byte Character Sets (DBCS)

This encoding uses both 8-bit and 16-bit encoding and usually referred to as multi-byte encoding. This is used mostly in CJK environments and uses leadbytes and trailbytes to map characters outside the 256 space. Most of the commonly used encoding schemes from SBCS and DBCS are adopted and integrated into Unicode.

Unicode

Unicode is a 16-bit character set that encompasses many characters used in general text interchange throughout the world. It contains 65,536 possible code points of which a third is still unassigned. Unlike other character encoding standards that assign character codes to both characters and glyphs, Unicode assigns character codes only to characters. Unicode is not a technology in itself. It allows for co-existence of many languages but it does not happen automatically. Tamil is a classic example of this. The number of characters defined in the Unicode table for Tamil is not enough to render

Tamil script. As such, in a font that is capable of rendering Tamil, the set of glyphs is greater than the number of Tamil characters defined in the Unicode table. Algorithms that specifically understand the mapping of defined characters to their associated glyphs need to be present in the system for it to fully render Tamil script with Unicode.

Design goals of Unicode

The original design goals of Unicode as defined in the Unicode Standard version 2.0 are as follows :

a. Universal – The repertoire must be large enough to encompass all characters that were likely to be used in general text interchange, including those in major international, national, and industry character sets.

b. Efficient – Plain text, composed of a sequence of fixed width characters, provides an extremely useful model because it is simple to parse: software does not have to maintain state, look for special escape sequences, or search forward or backward through text to identify characters.

c. Uniform – fixed character code allows efficient sorting, searching, display, and editing of text.

d. Unambiguous – Any given 16-bit value always represent the same character.

Implementing Tamil with Unicode

The Tamil character block in Unicode sits in the range of U+0B80->U+0BFF (which, in decimal, is 2944 -> 3071; 128 locations). The table below shows just the Independent vowels. The complete table of defined characters can be found in the Unicode Standard, Version 2.0.

Code Space	Character	Name
0B85	அ	TAMIL LETTER A
0B86	ஆ	TAMIL LETTER AA
0B87	இ	TAMIL LETTER I
0B88	ஈ	TAMIL LETTER II
0B89	உ	TAMIL LETTER U

0B8A	ஊ	TAMIL LETTER UU
0B8E	எ	TAMIL LETTER E
0B8F	ஏ	TAMIL LETTER EE
0B90	ஐ	TAMIL LETTER AI
0B92	ஒ	TAMIL LETTER O
0B93	ஓ	TAMIL LETTER OO
0B94	ஔ	TAMIL LETTER AU

The English names are defined in Unicode for each character and they are usually preceded with ‘TAMIL LETTER’ or ‘TAMIL VOWEL SIGN’. For example TAMIL LETTER KA refers to and TAMIL VOWEL SIGN E refers to -.

Memory Representation (Defined Characters)			Display (Glyph Table)
க	ி	→	கி
க	ீ	→	கீ
க	ு	→	கு
க	ூ	→	கூ
க	ை	→	கை
க	ோ	→	கொ

The number of characters defined in the table are insufficient to render Tamil script completely. Only the independent vowels (உயிர் எழுத்துகள்), Consonants (அகரம் ஏறிய உயிர்மெய் எழுத்துகள்), dependent vowel signs (modifiers) and Tamil numerals are defined. To be able to render Tamil script, the text processing system may map character sequences to their appropriate glyphs. The Unicode Standard version 2.0

deals with all the various combinations for Tamil and as such only a few are mentioned above.

Glyphs to render composite character sequences can be stored in a glyph table. From these examples, we can see that there is a considerable amount of interaction between character tables and glyph tables. The composition and layout process spans across both the tables. The presentation of glyphs based on the character sequence requires three primary operations :

a. selecting the glyph representations needed to display the character sequence.

b. assigning positions to the glyph shapes.

c. imaging the glyph shapes on screen or printer.

Glyph selection is the process of selecting (possibly through several interations) the most appropriate glyph identifier or combination of glyph identifiers to render a coded character or composite sequence of coded characters. Deleting of text may take the reverse process.

Using Tamil on shrink-wrapped software

With the complexities around implementing Tamil scripts with Unicode, it will not be possible to incorporate Unicode based Tamil text into off-the-shelf applications unless Tamil text handling capability is built into them. Even internationalised versions of these applications are written for specific environments and as such may not implement all the languages defined in Unicode. (It is also not a requirement for any application to implement all the code sets to be Unicode compliant).

In addition, the operating system should provide support in the input method as well as at the file system level in order to have applications that are Unicode compliant. Most major operating system vendors have announced support for Unicode and some have implemented it. But not all of them are expected to implement support for Tamil.

Developing “Tamil Aware” Applications with Unicode

It is possible for new applications to be developed for Tamil based on Unicode today. However, attention should be given to all text handling functions so that they do not break the 16-bit character (also known as wide character) rules. One simple example of

this will be not to assume that a byte is always equal to a character and not to perform pointer arithmetic as done with 8-bits. The advantage of developing on Unicode is the application can be easily moved around for other environments.

Non-graphical devices

From the sections above, it can be seen that Unicode requires a complex character to glyph substitution processes. As such it is more suited for sophisticated high-end desktops that has graphical displays and the capability of storing and rendering fonts from disk or memory.

Character rendering on non-graphical terminals are usually done through firmware (i.e. implemented in hardware). These devices do provide very limited banks of memory to load and render user defined fonts. Currently available devices are designed for simple text-entry and point of sale applications. Most of these just implement plain ASCII and fill up the extended space with line drawing characters.

Although graphical displays are taking over character terminals in most areas, the use of character terminals cannot be discarded in this part of the world. Especially in high volume areas where graphics devices cost substantially higher than character terminals.

As such a one-to-one (character to glyph) encoding is still required for Tamil. Though there are a few available, these efforts need to be synchronised and the standard made completely open for anyone to use without any kind of legal binding.

Conclusion

Unicode is where the future is. Although there aren’t that many environments that are Unicode ready today, they will be within the next few revisions. There will also be environments that will not move to Unicode. To survive both these environments, we need to define a single-byte character set for Tamil that fits into the most commonly used 256 code space while we develop software for the future. This will ensure that the usage of off-the-shelf shrink-wrapped software can be continued; only in a more standardised and consistent manner.

References

1. The Unicode Consortium, The Unicode Standard, Version 2.0, Addison-Wesley, Reading, MA, 1996.

2. ISO/IEC 15285: 199x, Working draft for An operational model for characters and glyphs, Version 9, January, 1997.
3. David Meltzer, Character Sets and Codepages, MS Typography, 1995.
4. Microsoft, TrueType Open Font Specification, version 0.1, March 1995.
Speakers'/Authors' Profile

Name : Muthu Nedumaran Email : muthu@murasu.com
Occupation : Technical Marketing Manager, SunSoft, Sun Microsystems Asia South region.

Tamil Computing Experience :Over 11 years. Developed the first Tamil interface on a PC with hardware modifications and assembler level device drivers in 1986. Presented a working system at the 6th International Conference on Tamil Studies in K.L., 1987. Developed MURASU range of Tamil desktop publishing interfaces for Windows and Unix. Developed MURSU Anjal in Jan 1995 – the most widely used Tamil email interface (over 35,000 users to date). Co-Founded Tamil.Net – Largest Internet mailing list with Tamil email exchange (over 300 subscribers). MURASU is use by almost all Tamil publications published in Malaysia and Singapore.

IT Experience : Over 12 years in IT. Currently with Sun Microsystems as regional technical marketing manager. Responsible for Internet/Intranet related software technologies and solutions. Awarded International Systems Engineer of the Year for 1996 by Sun Microsystems Inc. in Sydney, Australia.

Other Activities :Life member, International Association for Tamil Research, Malaysia.

ISCII And Tamil - A Perspective

N. Anbarasan
Applesoft, Bangalore, India
<anbu.arasan@axcess.net.in>
URL: http://www.irdu.nus.edu.sg/tamilweb/tamilnet97/paper/html/anbarasan.html

---

ABSTRACT

The urge for having a new Standard is arising from the usage of Computers to non-word processing requirements. The major segment where computers could be used massively is the Government sector, where, Indian languages (Tamil) could be used as medium in which databases could be maintained for various applications (requirements). The linguistic research Institutes also find interest in using computers for the linguistic study of the language. There has been a requirement to modify the existing ISCII Standard as it lacks proper sorting, indexing and character Identification in analysis. The purpose of this paper is to reflect the deficiencies in the draft ISCII Standard.

This paper brings out the suitable modifications in the draft Standard. It mainly focuses on the codification of the basic characters of Tamil to overcome the problems.

The Standardisation committee set up by the Tamil Nadu Government may take these issues to DoE and persuade them to rectify the deficiencies. If these views are not presented properly and at the earliest possible opportunity then the TAMIL language is likely to suffer from its setbacks and may be isolated from the Indian languages.

INTRODUCTION

All the Indian languages are believed to have been originated from the ancient Brahmi Script. All the Indian languages characters have varying shapes and forms, but have the common phonetic nature in sound, which is the basic for ISCII. The department of official languages and the department of electronics have been evolving Standards for character codes and keyboarding, which could cater to all Indian languages.

Even though these Standards are meant for all Indian languages, there is some discrepancies in the Standards which are not accommodating Tamil in full.

The current approach of DoE to have only Standard for Indian languages characters, leaving the display rendering and keyboarding mechanism to developers, seems to be a correct approach.

TAMIL SCRIPT

Initially, Tamil was used as merely sound to convey the feelings (unarchigal), and to communicate with each other. Later on a script (eluthu) is used to represent the sound. The script, after having undergone various changes at various stages (Kodugal, Pada eluthukkal, Vatteluthukkal, Chadhura eluthukkal) and is available in its present form. Today, we are discussing how best Tamil can be represented for the computer media. Again, the Tamil sounds are being coded not the script (eluthu).

Tamil Alphabets :

Tamil language has (30) basic characters (sounds) and is denoted by 12 vowels (uyir eluthukkal) and 18 consonants (mei eluthukkal).

Vowels : a, A, i, I, u, U, e, E, ai, o, O, au

Consonants : k, ng, c, nj, t, n, th, N, p, m, y, r, l, v, zh, L, R, nn

Composite consonants (uyirmei eluthukkal) are formed when vowel and consonants join together.

For example :

ka = k + a , ki = k + i etc.

Note the order, vowels come after consonants and combine with consonants to form composite consonant. When a vowel comes after a consonant, it always joins the consonant and it is represented by a auxiliary sign called vowel sign. This way, the combination of 12 vowels and 18 consonants form 216 composite consonants.

Tamil script is being taught, in schools based on this method only.

Number system :

Tamil number system is not same as English number system. Number '0' is not having any separate symbol. Numbers like 10,100,1000 etc., are having their own distinctive symbols. In practice, for e.g., in Government calendars we see these numbers are used as English numbers.

Sorting and Indexing :

Sorting and Indexing is one of the basic necessities of the database management system. Let me explain in detail, so as to give the impact on the importance of sorting and indexing. For example, we have to release the list of candidates of some examinations. We have to list the names in the alphabetical order. This helps to locate a name very easily. This type of sorting is used in real life applications.

Tamil words and names are sorted using the Tamil alphabetical order. Therefore, it is important not to alter this ordering.

ISCII STANDARD

A Brief History :

For the past few years the Department of Electronics (DoE) is sponsoring various projects using ISCII-91. Based on the requests for a revision in ISCII-91 from different developers, the DoE setup a committee in November 1996 to look into the problems faced by using the present ISCII-91 and to recommend the necessary revisions. The draft copy of ISCII - 97 is the outcome of the recommendations of this Committee.

Observation :

The proposed ISCII code is based on the ANSI Standard (please note that there are differences in the ASCII and ANSI Standards). Windows based software follows ANSI while DOS follows ASCII with extended characters (called graphics characters).

The Phonetic nature of Indian languages :

Based on the phonetic nature of the Indian languages, a common alphabet code is made possible. All vowels, consonants, graphic signs, punctuation marks, special symbols and extenders are coded. It provides a unique (common) encoding for all Indian languages.

ISCII Coding :

The revised Indian Standard Code for Information Interchange (ISCII) is a common encoding for all Indian languages. Table 1 shows this encoding with ASCII.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
1	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2	SP	!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
3	0/0	1/1	2/2	3/3	4/4	5/5	6/6	7/7	8/8	9/9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	DEL
8																
9																
A		Om	VM1	VM2	Chan-dra bindu	Anus-war	Visa-rg	Ayud-ham	a	A	I	I	u	U	ru	Ru
B	lu	lRu	e	E	ai	o	O	au	CM1	CM2	ka	Ka	ga	Ga	nga	cha
C	Cha	ja	Ja	nja	ta	Ta	da	Da	Na	tha	Tha	dha	Dha	na	pa	pha
D	ba	bha	ma	ya	ra	la	La	va	Sha	sha	sa	ha	dot	avag-rah	vira-m	hala-nt
E	add-ak															
F																

The ISCII code represents or contains only the basic alphabets. All syllables are formed through the combination of these basic characters. The rendering of the shape of a character is the process of the rendering software.

The ISCII has provision for future expansion in terms of inclusion of new language or new vowels or consonants by providing special codes (VMI and VML for vowels CMI and CML for consonants). This gives extensibility to encode newer Indian languages that may get official recognition in future. With these special codes, the ISCII can be extended to support 12x3 = 36 vowels and 34x3 = 102 consonants.

Study on the ISCII :

When we study the Standard in detail, we see the negligent care taken for Tamil. It puts one to think that Tamil is not considered as basic language, rather, it tries to accommodate the language as per the phonetic order instead of its own position. There exists some non-order of alphabets in general, which affects all Indian languages.

Ordering of alphabets :

The ordering of basic syllables itself has to be changed to have proper sorting. The combination of Anuswar and Visarg with consonants always comes after the vowel and the consonant combinations. In the Standard, as Visarg and Anuswar is having higher precedence than the vowels itself, the sorted list will have the combination of Visarg and Anuswar with consonants in the beginning rather than at the end. The new order could be vowels, graphic signs, consonants and special symbols.

How Tamil is affected :

The coding for Tamil is not as per the Tamil alphabetical order. The coding is affected just because of consideration of Sanskrit as a primary language. The Committee has tried to code the Tamil sounds as per the Sanskrit sounds. If you observe the ISCII table, surprisingly you will not find three Tamil characters. They are L, R, nn. These characters are supposed to be derived using other characters (l, r, n) with consonant extenders (special codes - for coding variations in consonant sounds differing in the pronunciation from Sanskrit). One can see that these characters were not considered as basic consonants but representation of sound borrowed from other languages and hence ordered as per the phonetic sound of Sanskrit. This not only affects the sorting, but also linguistic analysis of the language.

These characters have to be restored in its original position to have proper sorting and indexing.

Variants of varg consonants :

When we observe the ISCII code, we see that the variants of primary varg consonants (k, c, t, th, p) are not coded in Tamil. In Tamil, even though all the varg consonants are represented by a single consonant in written form, the phonetic variance is retained. As the ISCII is phonetic based, and is a common code for all Indian languages facilitating Transliteration. I suggest to have codes for all the left out varg consonants.

Problems in transliteration :

One of the good feature of ISCII is Transliteration. Transliteration is also affected by the ISCII code for he reasons mentioned below :-

1.In Hindi or in Kannada the character Anuswar is used in place of 'ng', 'n' Tamil equivalent characters.

1.As the variants of varg characters are not considered for Tamil, when one transliterates from other language, say Hindi, those variants will be missing in the transliterated data.

1.To have transliteration amongst Tamil and other Indian languages, we have to follow some grammars, because of the same reason.

Hence, it is not possible to achieve at least a simple minded 'one to one' transliteration.

GENERAL DISCUSSION ON ISCII

In the interest of better understanding of the ISCII, let me discuss in more detail. ISCII is meant for adaptation of Indian languages on computers in general. Unfortunately, even for English the standards are at variance for different platforms.

The more popular and widely used IBM PC range of computers use different operating system, such as DOS and windows 95 (win3.x is not an operating system by itself, it relies on DOS). As, DOS being the living system, and continue to be so for some more years, our ISCII also have to cater to DOS.

We can't use the proposed ISCII in PCs using DOS, because of the difference in extended characters of ASCII and ANSI. Therefore it needs appropriate arrangement of ISCII for ASCII (with extended graphics characters). As a software developer for Indian languages I have observed few things while implementing ISCII-91 in my software. The different software running even in the same Operating System do not accept all the character codes. Even if itaccepts, its interpretation will be different. Therefore , a study on the fairly good number of software will help us in deciding the codes.

Multilingual System :

Indian language numbers are coded in the place of English numerals. There is no code defined for language numerals. Now a days we see the requirement of multilingual software or at least trilingual where two Indian languages will have to coexist along with English. This coding could be implemented only in systems like windows. Whereas, it is not possible to have multilingual facility and different number system in systems like DOS and DOS based application software.

Compatibility with DOS :

Applications software developed for DOS in text mode relies heavily on graphics characters to have good looking screen designs. And also all leading software use graphics characters. Therefore, we can't have a common ISCII code for both ANSI and ASCII, and hence I would like to suggest that we have a different coding for ASCII also.

CONCLUSION

I request the Department of Tamil and Culture and the Tamil Nadu Government Standardising Committee to look into this subject and forward necessary changes to DoE.

In the interest of Tamil and Tamil people, I request the DoE to reconsider the draft copy and amend the required changes. If changes are not effected in the Standard there is every possibility that the Standard will remain just like a Standard, not used by Tamils. By accommodating the changes the ISCII is going to accommodate Tamil in the mainstream. If not, it may construe that Tamil language and its cherished richness are denied the just consideration.

BIOGRAPHY

Name : N. ANBARASAN
Designation : Chief Executive Officer
Organisation : APPLESOFT
mail address : anbu.arasan@axcess.net.in
Address for Paper Mail : No.39, 1st Cross, 1st Main, Shivanagar,
W.O.C. Road, BANGALORE - 560 010.
Telephone : 080 3357167 (Office)
080 3424765 (Residence)

Technology Developed:
(1) A graphical interface for DOS to have Indian languages on DOS.
1.Anti-aliased fonts for computer and TV medium to have
crisp and clear display.

Software Developed :

Developed a series of software to cater to various Indian languages requirements, as listed below.

SURABHI - 'Software only' solution for all Indian languages on DOS for text based software.
SIP - SURABHI Inscript Processor, a bilingual user friendly Inscript Word Processor having Wordstar compatible commands.
SUBASE VER 2.00 - Bilingual, general purpose database management software. It is software only solution.
SURABHI GEM - Bilingual and Multilingual software working on DOS, Ventura.
SURABHI SDK - Software Development Kit, to develop any bilingual Indian language software.
SURABHI UTILS - For day to day needs of the computer users.
SURABHI PRO - An interface software for Windows and Windows 95 to have all the facilities available.
AKSHARAM - Regional language learning Tutor.
JANANI - Interactive user friendly software to learn vernacular typing.

Selection and Standardization of Tamil Keyboard Layouts - Recommendations of Tamil Nadu Standardization Committee

Prof S. Kuppuswami and Mr. V. Prasanna Venkatesan
Department of Computer Science, Pondicherry University, Pondicherry, India

1. Introduction

Language is an essential tool for human life. Refinements in languages are necessary to make them more suitable for use with modern gadgets in order to provide sophisticated life for the people. The process of refinement has taken place continuously in may languages throughout the world.

Tamil is one of the most popular ancient languages used by a large community in India and also a considerable population all over the world. Right from its inception continuous changes have been made in Tamil to adapt to the linguistic, cultural and environmental changes.

The evolution in computer technology and its increased applications demand the adaptation of Tamil in computers. Towards this objective, the Department of Computer Science, Pondicherry University has submitted a research proposal to the Tamilnadu State Council for Higher Education (TNSCHE) to develop a Tamil Computer.

2. Formation of Standardization Committee

Based on the proposal submitted by the Pondicherry University, the Vice-Chairman of TNSCHE made his recommendations to the Government of Tamilnadu to constitute a committee to address the following issues and make suitable recommendations for developing Tamil Computer.

- Tamil character set
- Coding scheme
- Technical words
- Keyboard layout

Accepting the recommendations, the Government of Tamilnadu constituted a committee consisting of the following members.

1) Dr. M. Anandhakrishnan Chairman
Vice-Chairman
Tamilnadu State Council for Higher Education, Chennai

2) Dr. S. Kuppuswami Member
Professor & Head of Computer Science
Pondicherry University , Pondicherry

3) Mr. S. Rangarajan (Sujatha) Member
Writer, Chennai

4) Mr. N. Govindasamy Member
National Institute of Education
Nanyang Technological University , Singapore

5) Mr Bhakaran
Head of Computer Science Department Member
Tamil University , Thanjavour, Tamil Nadu

The Committee decided to go into the various issues mentioned above on a priority basis. As the standardization of Tamil keyboard layout is a primary issue, the committee has first taken up this work.

3. Categorization of Tamil Keyboard Layouts

At present lot of Tamil software have been made available mainly for word processing applications. Each one adapts its own keyboard layout. This necessitates the user to learn different keyboard layouts and thus complicates the keying-in process. This is due to the non-availability of standard Tamil keyboard layout.

In order to select and standardize a Tamil keyboard layout, the committee has decided to categorize the existing Tamil keyboard layouts and analyze them.
Tamil keyboard layouts proposed by the various software developers and researchers have been collected and categorized into four groups. They are presented below.

a) Phonetic keyboard layouts

Keyboard layouts designed based on the phonemes and frequency of usage of Tamil characters are classified as Phonetic keyboard layouts. The following layouts fall under this category.

- Kanian
- Nalinam
- DOE
- Krishnamoorthy's

b) Typewriter-like keyboard layouts

Layouts which follow the Tamil typewriter machine keyboard are classified as Typewriter-like keyboard layouts. Some of them are

- Typewriter
- Inscript
- Annai

c) Romanized keyboard layouts

In Romanized keyboard layouts, mapping of the Tamil characters to the corresponding English characters is done on transliteration basis. The following layouts are the examples of this category.

- PONN
- Murasu
- Yarzhan
- Chellapan's

d) Others

The keyboard layouts which do not fall under the above categories are grouped as Others. Some of them are

- Ventura
- GIST
- Deskset

4. Analysis of the Tamil Keyboard Layouts

An efficient keyboard layout has to meet the following requirements. Any keyboard which meets these requirements will have features like easy handling, simple learning and standard usage.

The keyboard layout should consume less number of key strokes.

In the keyboard layout, mapping of the characters to Strong and Weak key positions should be based on the frequency of usage.

The keyboard layout should have distribution of keys on the left side and right side such that the keys are used alternately by left and right hand, during the keying-in process.

An extensive analysis was carried out in three phases on the Tamil keyboard layouts, mentioned in the previous section, to determine their efficiency for keying-in of Tamil text. For this analysis, Tamil text were selected from various Tamil literature of different period. The objective, methodology and observations of the analysis carried out in three phases are presented below.

Phase I

Objective:
Identifying keyboard layouts which consume less number of key strokes.

Methodology:
In this phase, character count has been performed on each of the keyed-in Tamil text and it has been converted into number of actual key strokes corresponding to different keyboard layouts. The character count and number of actual key strokes are tabulated in Table - I.

Observations:
Phonetic keyboard layouts perform better than the other three types of keyboards. Kanian, Nalinam and Krishnamoorthy's layouts lead in this type.

Typewriter-like keyboard layouts closely follow phonetic type in this analysis.

Romanized keyboard layouts come in third place. Among them Yarzhan and PONN layouts lead the list.

Other types of keyboard layouts get the last position in the list.

Phase II

Objective:
Identifying keyboard layouts which have high frequency of usage for strong fingers and low frequency of usage for weak fingers.

Methodology:
The second phase analysis is done using the same text used for the first phase. The frequency of usage of each key position is calculated from which the usage of strong and weak fingers, in percentage, are determined. The results thus obtained are given in Table - II.

Observations:
In this analysis also Phonetic keyboard layouts occupy the first position
The Typewriter-like keyboard layouts and the Romanized keyboard layouts closely follow the phonetic layouts.
Other types of keyboard layouts showed varied performance.

Phase III

Objective:
Identifying keyboard layouts which have keys distributed on the left side and right side of the keyboards such that the keys are used alternately by left and right hand.

Methodology:
The same text are used in this phase. The usage of each key position is calculated from which the toggling of left and right hand, in key strokes, are determined. The number of key strokes made without toggling are also determined. Table-III shows the results obtained.

Observations:
Phonetic keyboard layouts outperform all other layouts. Amongst them Kanian shows better results.

Typewriter-like keyboard layouts occupy second position.

Romanised and Other type keyboard layouts show varied performance.

5. Conclusion

Based on the results obtained from the analysis, the Tamil Computer Standardization Committee recommends the following for selecting and standardizing keyboard layouts for Tamil computer.

Phonetic keyboard layouts give better performance in all the three phases of analysis. Hence it has been decided to select the best phonetic keyboard layout and that may be refined further.

As large number of typewriter trained personnel are readily available, it is also felt that the best Typewriter-like keyboard layout can also be selected and improved.

For the Tamil people living all over the world, who are using English keyboard to input the Tamil text, Romanized keyboard layout is essential. Therefore, one keyboard layout can be selected from this category and refined.

Hence, umbrella standard of keyboard layouts have to be made available from each category of keyboards. The work is in progress in this direction.

About the Author

Prof. S. Kuppuswami
Head Department of Computer Science
Pondicherry University
Pondicherry - 605014, India

Mr. V. Prasanna Venkatesan
Lecturer,Department of Computer Science
Pondicherry University
Pondicherry - 605014, India

Sri Lankan Experience of Development of Tamil Input/Display Methods

ST Nandasara
stn@sfc.wide.ad.jp, stn@ict.cmb.ac.lk
Institute of Computer Technology, University of Colombo, Sri Lanka

Abstract

Sri Lanka is a multi-racial society comprising of a 74% Sinhala-speaking population and a 18% Tamil-speaking population. Sinhala, Tamil and English are all official languages of Sri Lanka and are extensively spoken throughout this country of 18 million people.

In 1989, we launched our Sinhala/Tamil Word processor for DOS environment and initially it was available only in text based environment. This development of the interface for DOS was two-stage process. The first stage was to develop the BIOS part and the second was development of Text Based Word processor including Keyboard Input method and the Screen Handler.

This experience was very much helpful for the later stage to development such Input systems for the Graphical User Interface (GUI) like Windows95 and now research are being place in Sri Lanka to adapt this method. As a result Tamil Input/Display methods are available for Internet users as well.

In July 1996, Sri Lanka launched its National Website (http://www.lk) initially consisting of information entirely in the English language, as it is thelanguage of commerce and government second language. Moreover, most people currently with Internet access and computer-literate are proficient inEnglish.

However, because of our population profile, it is absolutely essential that Internet data be made available in Sinhala and Tamil in order for the Internet to reach our masses.

In addition, keyboard input technology is being developed to allow users to download, search and create new data and build their own content in all threelanguages. As demand increases, this will encourage information providers to leverage on the technology to put up more quality information in all three languages so that Internet technology can take root in Sri Lanka.

Introduction

The Tamil script is a South Indian script. South Indian scripts are structurally related to the North Indian scripts, but they are used to write DravidianLanguages of Southern India and of Sri Lanka, which are genetically unrelated to the North Indian Languages such as Hindi, Bengali, and Gujarati. Theshapes of letters in the South Indian script are generally quite distinct from the shapes of letter in Devanagari and its related scripts. This is partly a result of the fact that the South Indian scripts were originally carved that square, block-like shapes.

The Tamil script is used to write the Tamil language of Tamil-Nadu State in India as well as minority languages such as Badaga. Tamil is also used in Sri Lanka, Singapore, and parts of Malaysia. The Tamil script has fewer consonants that the other Indian scripts. It also lacks conjunct consonant forms. Instead of conjunct consonant forms, the virama is normally fully depicted in Tamil text.

Nature of the Tamil Language

The Tamil alphabet (Tamil-alphabet ::= <Vowels><Consonants><Consonant -modifiers>) consists of 48 symbols: 22 consonants, 12 vowels and 13 consonant modifies as follows:Table 1 A summary of the Tamil Letters

Consonants

Consonant modifiers in Tamil, which are graphical signs, used in conjunction with Tamil consonants. These consonant modifies can occur in left, right and top of any Tamil consonants. However no consonants modifies can occur bottom of any consonants where as Sinhala it occurs.

Special Characters

Tamil AU LENGTH MARK (??) is provided as an encoding for the right side of the surroundrant (or two-part) vowel sign AU (È??). Note that the Tamil vowel sign AU LENGTH mark (??) is not the Tamil Letter LLA (?).

Development early 1990s

The Institute of Computer Technology, University of Colombo Started development of Sinhala/Tamil Word Processor for Personal Computer under the Disk

Operating System (DOS). However, this word processor system is not capable of mixing all tree language together in one document due to the limitation of DOS operating system. We have designed a bilingual font set for the display of both Tamil and English or Sinhala and English simultaneously. This was done making use of the upper extended ASCII character range for the Sinhala and Tamil characters, while retaining the basic alphabet and punctuation marks in the lower ASCII range. Language can be selected by toggling the key combination when ever is required.

In 1991, this was demonstrated in the Annual Computer Society Conference in Colombo and it allowed for the most of the e-mail users in the world to be send or to read messages in both languages simultaneously English and Tamil or English and Sinhala. System will be recognized and displayed Tamil or Sinhala font correctly when they occur, without having to change font set.

For this development, at that time we recognized 12 vowels (£, ?, ¥, ?, §, ¨, ©, ª, «, ¬, ? and ®) and 18 consonants (¯, °, ±, ?, ´, µ, ¶, ·, ¸, ?, º, », ?, ?, ?, ¿, À and Á) in Tamil Language. These 30 sounds are the initial sounds of the Tamil Language and the basic for the Tamil alphabet, where as the new Tamil character set consists of more consonants as shown in Table 1. The 18 consonants joined with 12 consonant modifiers (ÿÇ, ÿÈ, ÿÉ, ÿÊ, ÿË, Ìÿ, Íÿ, Îÿ, ÌÿÇ, ÍÿÇ, Îÿ¿ and ÿÆ) form the remaining 216 glyphs for the language. The special character ? (SRI) were added to the Tamil Character set, which was used in Sri Lanka for some time. Table 2 below shows the Total character set and Table 3 shows the ASCII character map for the English-Tamil font set.

Table 2 - Total Tamil Character Set

?

Table 3 - ASCII character map for the English-Tamil font set

With this code table we make use of the over print capability (without moving cursor to the next location) and specially design consonant modifiers to keep new glyph together to generate new glyph, that is not found in the above ASCII table. For example, glyphs of row number 3 of the Table 2 can be formed with the corresponding consonant with its modifier ÿÈ together side-by-side. Glyphs of row number 4 and 13 can be generated by over printing the consonant modifier ÿÉ and ÿÆ. Some of the glyphs such as è and Ú were formed with the combination of ° and ÿÄ, and ? and ÿÃ respectively.

Rendering of Tamil Script

The South Indic scripts function in much the same way as Devanagari, with the additional feature of two-part vowels. As in the Devanagari example, the words "TAMIL LETTER" and "TAMIL VOWEL SIGN" will be omitted where this does not cause ambiguity.

It is important to emphasise that in a font that is capable of rendering Tamil, the set of glyphs is greater than the number of Tamil Characters.

It is evident that the Tamil character set consisting of vowels, consonants and consonant modifies have clear differences, mainly with respect to the size of the characters. Some characters are much bigger than the others. Their shapes also differ. Although the basic shape of the character is curved, some parts are strait lines.

Unlike in Roman Scripts, most of the Indic language consonant modifies could be positioned at different locations around the consonants. These consonants modifies for Tamil can be classified in to three groups as follows.

<Consonant-modifiers> ::= <Left-modifiers><Right-modifiers><Upper-modifiers>

In the Tamil language, combinations of consonants and consonant modifiers produce different phonetic sounds. For example, the combination of the consonant (KA) and consonant modifies given in Table 1 produce 12 different phonetic sound for the character (KA). See Table 7 for these combinations.

In Tamil, it is important to emphasise that in a font that is capable of rendering combinations of Tamil script, the set of glyphs is greater than the number of Tamil Characters (See Table 4). However, the total number is fit in to 25x13-matrix and letter ? it is equivalent to 326 glyphs and this including the vowels, consonants and consonant modifies.

Vowel Reordering

As shown in Table 5, the following vowels are always reordered in front of the previous consonant cluster. The similar behavior is available in Sinhala as well (See Table 6).

Sinhala = ïÿ

Tamil = Æ? Ç? È?

Table 5 - Vowel Re-ordering in Tamil
Key-in
Memory Representation

Table 6 - Vowel Re-ordering in Sinhala

Table 4 - Tamil Glyphs

Ligatures

The following examples illustrate the range of ligatures available in Tamil. These changes take place after vowel reordering and vowel splitting

1. The vowel "?" optionally legates with £, ©, or ± on its left.
   £ + ? -
   © + ? -
   ± + ? ˜

Since this process takes place after reordering and splitting, the following ligatures may also occur:

Separate Vowels
£ + Æ? + ? Æ-
£ + Ç? + ? Ç-
© + Æ? + ? Æ-
© + Ç? + ? Ç-
± + Æ? + ? Æ˜
± + Ç? + ? Ç˜

2. The vowel sign "?¿" and "?À" form ligature with "Ÿ" on their left.
   Ÿ + ?¿ =
   Ÿ + ?À T

These vowels often change shapes or position slightly to link up with the appropriate shape of the consonant on their left:

? + ?¿ G
? + ?À c

1.The vowel signs "?Ã|Á" and "? Ä|Â" typically change form of legate.



4. To the right of œ, ', ', ", or Å these forms have a spacing form.

œ + ?Ã œÁ
œ + ? Ä œÂ
'+ ?Ã 'Á
'+ ?Ä 'Â
' + ?Ã 'Á
' + ? Ä 'Â
" + ?Ã "Á
" + ? Ä "Â
Å + ?Ã ÅÁ
Å + ? Ä ÅÂ

1.The vowel sign "È?" changes to "Å?" to the left of "£", "©", "?", or "?".

È? + £ ™
È? + © ?
È? + ? ›
È? + ? œ

Remember that this change takes place after the vowel reordering; in the first example, the vowel "È?" follows "£" in the memory
representation. After vowel reordering, it is on the left of "£", and thus changes form. The complete process is

£ + È? È? + £ ™

Tamil Character Font Set

In the implementation, we have designed a font set for the display of Tamil (See Table 7). This was done by making use of the upper extended ASCII character range for the Tamil character, while retaining the basic English alphabet and punctuation in the lower ASCII range. This will allow the most of the all Tamil glyphs to displayed correctly, some using precompiled glyphs or use of the kerning feature built into the True-Type Font Technology to combined two Tamil characters into a new character glyph not found in the Font Table. With the combination of two or more Tamil character or Vowels set to give a more complex glyph, we can then include the entire Tamil character set within one single font.

Table 7 - Tamil Font Table

Tamil Keyboard layout Design
Figure 1 - Tamil keyboard Layout

Conclusion

Sri Lanka National Web Site was designed to provide information in Sinhala, Tamil and English. The development of Tamil input system is equally important to the Sri Lankan society to allow for Trilingual Web Display, E-mail Interchange, Keyboard Input and other essential Internet functionalities. Trilingual Web Site allows users to view same piece of information in their preferred language, which could be English, Sinhala or Tamil. It also allows the user to search the database using their preferred language.

References

1.Govindasamy, n (1989). Keyboard for Tamil Computer, by Naa Govindasamy, 7th International Conference of Tamil Studies Seminar Proceedings, Mauritius, December 1989.

2.Sri Lanka Standard for the Sinhala Character Code for Information interchange, Sri Lanka Standard institute, Sri Lanka, 1996.

3.Report of the Committee for Standardization of Keyboard layout for Indian Script Based Computer, Electronics-Information & Planning, Vol 14, No. 1 October 1986.

Speakers'/Authors' Profile

Authors : S.T. Nandasara (stn@sfc.wide.ad.jp)
Organizations : Institute of Computer Technology, University of Colombo,
PO Box 1490, Colombo, Sri Lanka
Point of Contact: S.T.Nandasara (stn@sfc.wide.ad.jp)
Tel: 941-503150 Fax: 941-587239
URL : http://www.ict.cmb.ac.lk

ST Nandasara

Standardization of Tamil Keyboard Layouts:
Recommendations of Tamil Nadu Standardization Committee

Tamil computer is a long felt dream of the Tamil Society. Many attempts were made to achieve this. But those attempts are limited due to lack of standards. To realize the Tamil Computer, Tamil Nadu Government constituted an expert committee, which is governed by Tamil Nadu State Council for Higher Education, Chennai. A separate history of the development of Tamil computers will be published later by the committee.

The Committee addressed various issues namely Tamil Character Set, Orthography, Keyboard layouts and Coding Scheme, and planned to provide standard solution.

As a first step, the Committee has decided to keep the existing Tamil Character Set, and Orthography as the standard one. Suggestions have also been made to bring back the "Before Periyar modification letters," like y s etc, which will be very useful in character recognization applications and scanning-in older Tamil books.

As a next step the Committee has taken up the standardization of Tamil keyboard layouts, that is the key issue. It has collected more than fifteen keyboard layouts which are in use, and catagorized them as,

1. Phonetic
2. Typewriter-like
3. Transliterate (Romanized).

Then, those layouts were analyzed and ranked for different key characteristics, like minimal key strokes, fingers' usage, etc. The Phonetic layouts took the first position, and the Typewriter-like layouts and Transliterate layouts are placed in second and third position respectively.

Based on the above excercise, the Committee has standardized Keyboard Layouts in each of the above category as given below.

Tamil Nadu Government's Phonetic Layout :-

தமிழக அரசின் ஒலியன் விசைப்பலகை :-

Since this layout is an efficient one, it has to be adopted for future Tamil computer usages. The layout comprises of the twelve vowels, the first eighteen vowel-consonants (mfuBkwpa bka;) and the kirandam letters.

phonetic.jpg (19232 bytes)

The following example shows the key combination to generate the consonants and vowel-consonants.

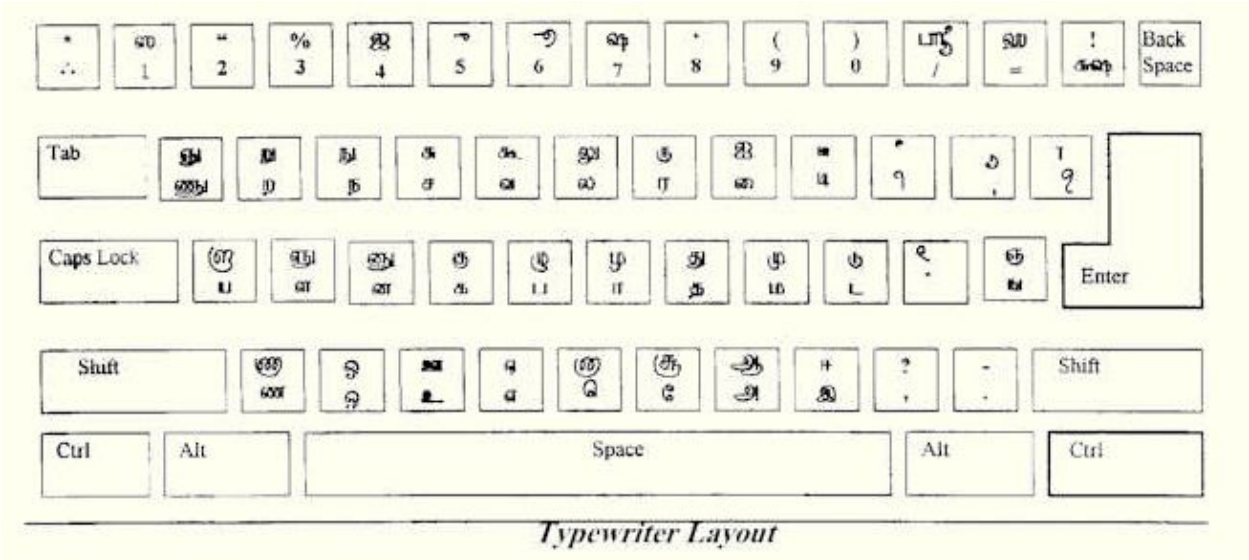
க + ; = க்
க + ஆ = கா
க + n = கி
க + N = கீ
க + உ = கு
க + ஊ = கூ
க + எ = கெ
க + ஏ = கே
க + ஐ = கை
க + ஒ = கொ
க + ஓ = கோ
க + ஔ = கௌ

The Committee also decided to reserve one function key for inputting old style letters and another for speed-typing (The layout that has vowels and consonants only). These function keys should not clash with standard DTP functions.

Tamil Nadu Government's Typewriter layout :-

தமிழக அரசின் தட்டச்சு விசைப்பலகை :-

Considering the existance of large number of typewriter trained personnel, the typewriter layout is selected and is shown below.



Typewriter Layout

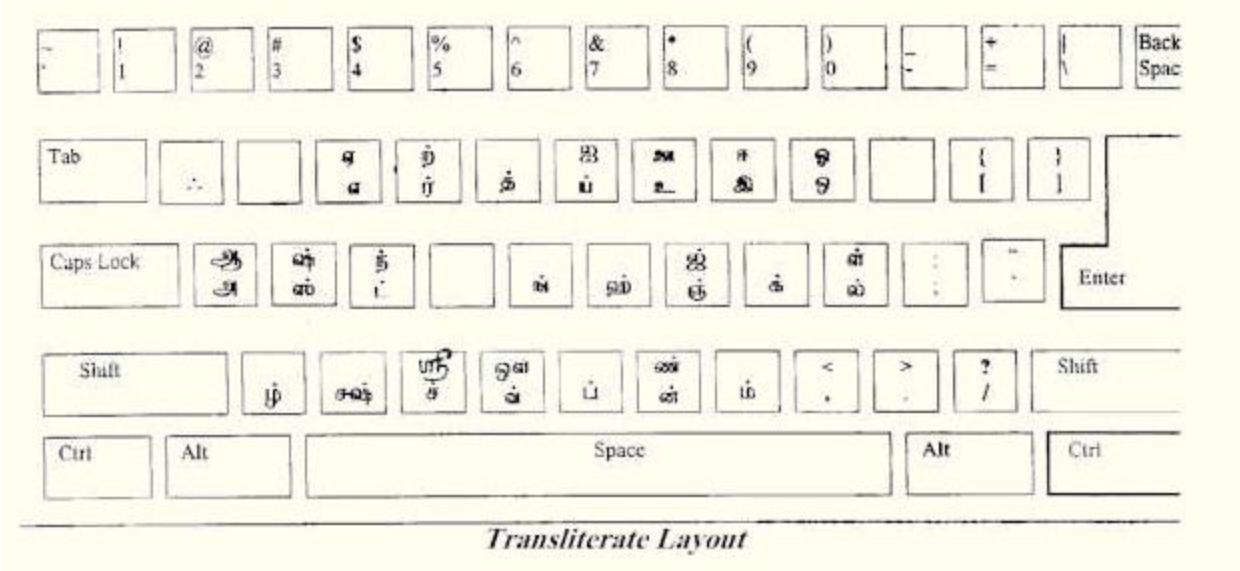
The following example shows the key combination to generate the consonants and vowel-consonants.

க + ; = க்
க + h = கா
க + p = கி
க + P = கீ
கு = கு
கூ = கூ
டி + க = கெ
டீ + க = கே
i + க = கை
டி + க + h = கொ
டீ + க + h = கோ
டி + க + ள = கௌ

Tamil Nadu Government's Transliterate layout :-

தமிழக அரசின் ஒலிமாற்றி விசைப்பலகை :-

For the convenience of users who are familiar with English keyboard, a standardized transliterate layout is given below. This layout is not suitable for voluminous data input.
trans.jpg (21691 bytes)



Transliterate Layout

The following example shows the key combination to generate the vowel-consonants.

க் + அ = க
க் + ஆ = கா
க் + n = கி
க் + N = கீ
க் + உ = கு
க் + ஊ = கூ
க் + எ = கெ
க் + ஏ = கே
\க் + ஐ = கை
க் + ஒ = கொ
க் + ஓ = கோ
க் + ஔ = கௌ

The committee also noticed the method used by various research organisations at home and abroad for transliteration of Tamil using only lowercase English alphabets with additional diacritical marks. This has to be studied with all its implications.

Comments on these keyboard layouts may be sent to any one of the following email addresses before 31st July, 1997.

1) Prof. M. Ananthakrishnan ananda@md2.vsnl.net.in
2) Prof. S. Kuppuswami swami@pondiuni.ren.nic.in
3) Mr. S. Rangarajan sujatha@md2.vsnl.net.in
4) Mr. Naa. Govindasamy

TAMILNET'97: Discussion Forum - A Summary Report

The 2 hour-long discussion session on the second day of the symposium came to a fruitful ending. The discussion panel (a total of 11 members) was led by Sujatha and included N. Anbarasan, S Kuppuswami, ST Nandasara, Kuppuswamy Kalyanasundaram, Malaan, Harold Schiffman, Vasu Renganathan, Muthelilan Murasu Nedumaran, Naa Govindasamy and Tan Tin Wee.

The topics discussed includes -

1. Increasing Tamil Content Worldwide
2. Keyboard Diversity
3. Encoding Standards
4. Standardization of Tamil Websites and WebPages
5. Work to be done/Time Frame
6. Next Symposium - TamilNet'98

Increasing Tamil Content WorldWide

A project similar to the Project Gutenberg being pursued in Germany, should be initiated for the Tamil language. This project should cover classical literay works, contemporary literary works, teaching and learning materials, search tools, compilation of bibliography and dictionary. It is hope that authors of these works will donate them for the Net. Ways should also be looked into for copyright protection etc. Mirror sites of these information will also help in the growth of such Tamil content on the Net to reach a wider audience.

Keyboard Diversity

The current wide range of keyboard layouts for Tamil input has led to much confusion and inefficient use. As such, the panel hopes that a set of umbrella standards will rectify the situation. This umbrella standard will hopefully be

1. Direct - Typewriter method
2. Interpreted - Phonetic+Romanized method

Of course, user preferences will play a major part in the setting of these umbrella standards. Open market and fair competition will encourage interchangeable keyboards.

Encoding Standards

The panel agreed that there should be a unified 8-bit character set that will co-exist with Unicode for the Tamil language. The TamilNadu Computer Standardization Committee will work with developers towards a unified 8-bit character set. To facilitate theinterchange of the myriad of 8-bit codes around, code conversion has to be provided for these codes. The use of Unicode as the
standard interchange code is proposed to coexist with the unified 8-bit character set for future advancement of Tamil information technology.

Standardization of Tamil Websites and WebPages

For the convenience of Web users surfing the Net and reading Tamil webpages, websites should maintain at least two formats - own coding format and the interchange code (be it Unicode or unified 8-bit code) Towards this end, the Tamil Nadu government will fund a universally and freely accessible 8-bit code standard font (with minimal sorting order problems and transliteration support)

Work to be done/Time Frame

The time limit for keyboard standardization is targetted to be 30 June 1997.
Internet software developers and researchers can submit their proposed 8-bit character set encoding to the TamilNadu Computer Standardization Committee for their review by the end of June 1997. Thereafter, the committee will based on these submissions prepare the first draft of the 8-bit character set encoding by July 1997. A RFC (Request For Comments) should be floated on the Internet for comments in 2-4 weeks time once the above-mentioned final draft is ready. A final standardization decision should be made by Jan/Feb 1998 given that the RFC document should have expired after 6 months (i.e. in Jan/Feb 1998). A submission of the final standard document to IETF or standardization body like ISO is possible then.

If you would like to submit a working proposal for the unified 8-bit encoding standard, you can contact Sujatha at sujatha@md2.vsnl.net.in or Dr S Kuppuswami at swami@pondiuni.ren.nic.in who are both in the TamilNadu Computer Standardization Committee.

Next Symposium - TamilNet'98

The next symposium is tentatively set to be held in early 1998 with the final date to be decided. The symposium will preferably be hosted in an acadamic institution in Chennai (Madras), Tamil Nadu, India. The organizing committee is to be formed by the TamilNadu Government. Commercial exhibition and sponsorship to the event is welcomed.