8. Technology Review

8.1 Font Technology

Standards play a very important role in making a product C-DAC, GIST along with experts from all over the Nation played a vital role in developing standards for Indian language technology products.

In a plain simple layman terms, following bare minimum things need to be addressed for enabling any system / device(s) with Indian languages.

The way you store.

The way you input.

The way you display.

For storing the data ISCII / Unicode encoding may be used and for inputting data INSCRIPT keyboard layout may be used.

The third important aspect of Indian languages – comprting i.e. the display part or more specifically the FONTS is addressed here.

Before proceeding to in-depth discussions on the Fonts & related technologies, some background on related topics is also presented here.

ISCII (Indian Standard Code for Information Interchange) – *The storage standard*

Prior to 1988, Indian language related activities were generally limited to specific language(s) and were independent work of various organizations. Thus making data interchange impossible. In such a scenario, it was important to have a common standard for coding Indian scripts.

In 1991, the Bureau of Indian Standards adopted the Indian Standard Code for Information Interchange (ISCII), the ISCII standard that was evolved by a standardization committee, of which C-DAC was a member, under Department of Electronics during 1986-88. The ISCII document is available as IS13194: 1991 from the BIS offices.

ISCII – 91 is the latest standard which was more refined version of ISCII – 88 standard. One of the major differences in ISCII –88 & ISCII –91 is the addition of Vedic support.

All C-DAC, GIST products are based on ISCII standards. Other than this, several companies developing products have used it and had solutions based on this representation. This has been made mandatory for the data being collected by organizations like: The Election Commission, and for projects such as Land Records Project, etc which are of National importance.

ISCII is an 8-bit coded character set for Indian Scripts. In this, the lower half (first 128 characters) is kept reserved for the 7-bit ASCII character set. This makes it possible to use Indian scripts along with existing English characters on computers with 8-bit character codes.

The ISCII code co-exists with the 7-bit ASCII code. It contains only the basic alphabets, for a particular script, arranged in a lexicographic order (that's according to the one found in most of the Indian dictionaries).

The ISCII code table is super-set of all the characters required in the ten Brahmi-based Indian scripts. All of the 10 scripts have lot of similarities. All these similarities are because of having the same evolution - the ancient Brahmi Script. This all resulted in having a common phonetic structure.

There are 15 officially recognized languages in India: Hindi, Marathi, Sanskrit, Punjabi, Gujarati, Oriya, Bengali, Assamese, Telugu, Kannada, Malayalam, Tamil, Urdu, Sindhi & Kashmiri.

Out of these Urdu, Sindhi & Kashmiri are primarily written in Perso-Arbic scripts (Naskh / Nastaleeq), but get in written in Devanagri too (Sindhi is also written in Gujarati Script).

As Perso-Arabic scripts have a different alphabet, a different standard is being proposed called as PASCII (Perso-Arabic Standard for Information Interchange) or "Hindustani".

Keeping the tradition, this standard is designed and developed by C-DAC, GIST as a part of TDIL funded "Resource Centre" project, This is also 8 bit standard & is published in TDIL journal for expert comments & opinion. For more details of this standards pl. refer the TDIL journal or visit look into Perso-Arabic page of C-DAC's website.

INSCRIPT Keyboard : *The inputting standard.*

INSCRIPT keyboard overlay contains characters required for all the Indian scripts as defined by the ISCII character set. The INSCRIPT keyboard is the most scientifically developed standard, which is based on the phonetic nature of the Indian languages. The scientific structure allows a very easy learning and also a person who knows typing in one Indian script can type in any other Indian script. It is devised on the fact - "*the way we pronounce the way we type*" which is unlike the typewriter layout, wherein it is based on the "*way you see the way you type*".

INSCRIPT keyboard is most suitable, since in the school / colleges, the phonetic structure of Indian languages is taught while reading / writing.

The third and the most essential part is the display which is of concern to the user rather than the developer. The user is unaware & need not know the nitty-gritty of how things works, but is more interested to see how things are displayed / conveyed. Fonts play a very vital role in communicating. Right font style at right time & right occasion may also become a matter of research.

Usage of Fonts

Various applications under MS-Windows, Mac, Unix, Solaris, Linux, etc.

Web based applications.

Caption / Character Generator, Teleprompter - broadcast equipment's.

Specially designed mono thick fonts have been used for greater legibility and readability on TV screen for subtitling purposes.

Fonts for 9 pin, 24 pin dot matrix printers, pagers, mobiles, PDA's have opened up a new area for printed and displayed word in Indian languages

Matching English fonts.

Fonts for LED / LCD based public information display systems.

PostScript Type –1, True type and bitmap fonts for high quality publishing and printing.

In English script, there is one to one correspondence between what you type, what you store & what you display. Also it is a linearly mapped according to the character set. It does not grow vertically like Devanagari or Arabic script. Which necessarily means that if 'A' is inputted from the keyboard then internally the code of 'A' which is 65 in ASCII is stored, while displaying, the Glyph / shape of the letter 'A' placed at the position 65 in the font is used for rendering.

Life is not as simple as English while using Indian languages. Due to the highly complex structure of the Indian languages, there is no one to one correspondence between what you store (character code) & what you display (font code). The relationship between characters and glyphs can be one-to-one or one-to-many or many-to-one.

Formation of multiple shapes because of nature of the scripts

Most of the scripts have a peculiarity that most of its characters can have multiple visual representations depending upon its position in the word. This can be seen in case of Arabic script wherein the character shape has four different forms depending upon whether the character is in beginning, middle, end or standalone which is shown below.

Name	Final	Middle	Begin	Standalone
Alif	ا			ا
Be	ب	ـبـ	بـ	ب
pe	پ	ـپـ	پـ	پ
te	ت	ـتـ	تـ	ت
Te	ٹ	ـٹـ	ٹـ	ٹ
se	ث	ـثـ	ثـ	ث
jeem	ج	ـجـ	جـ	ج
ce	چ	ـچـ	چـ	چ

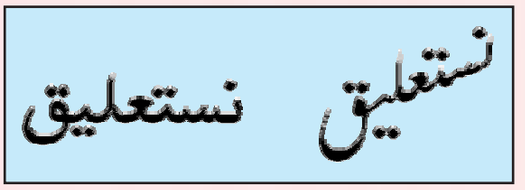
More or less the same is true with Devanagari script, depending upon whether or not a consonant is followed by halant so as to have half or full form. It also depends upon whether for proper visual representation long sized or short sized e-matra is used. Due to this the number of glyphs far exceeds

the number of characters in these scripts. And hence there is no one to one correspondence in the storage & fonts standard.

कि : Herein to display, a short in size e-matra is used to fit exactly on top of middle axis of letter Ka

क्यि : Herein to display, a long in size e-matra is used to fit exactly on top of the combination so as to retain the character aesthetics and meaning.

For e.g. the characters क, ◌ि, य & ◌ी are used to form the cluster क्यि



Script=Naskh Script=Nastaleeq

A Sample text in Naskh and Nastaleeq scripts

The common phonetic structure of Indian language plays an important role in coming up with a common character set. Being phonetic, each alphabet, in these languages, represents a particular sound. That means, in Indian languages, there is a direct correspondence between sounds and letters. Written as it is pronounced. (*phonetic alphabet)

All these are written at syllable level. A Syllable is a sound, unit of pronunciation having a vowel or diphthong sound optionally associated to one or more consonants. eg. "kya" is a single syllable. In devanagari, this is represented with a conjunct -"क्या"

ASCII (American Standard Code for Information Interchange) and ISCII (Indian Standard Code for Information Interchange)

ASCII for English characters has one to one correspondence. That means, to represent an English alphabet there is a corresponding single numeric character code in ASCII.

In ISCII, the formation of syllables is achieved through multiple character sequences. In Indian language scripts, a group of characters together form a syllable or a cluster

Syllable = Multiple character Sequence

eg. "*brahmaa*" (ब्रह्मा) has 2 syllables - '*bra*' (ब्र) and '*hmaa*' (ह्मा)

'*bra*' is a sequence of 3 character codes ब्र = ब + ◌् + र.

'*hmaa*' is a sequence of 4 character codes ह्मा = ह + ◌् + म + ◌ा.

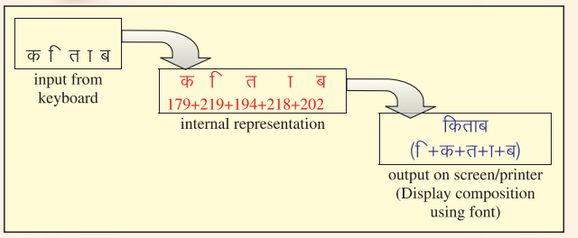
ISCII and the ISFOC (Indian Standard Font Code) coding

ISCII codes have nothing to do with fonts or for that matter displaying of a script in a particular style, because ISCII is an encoding standard (data storage) standard. A text in ISCII may be interpreted to display the text using different fonts for the same script or different scripts.

This mapping will require a converter, which will do multi-byte syllable mapping into glyphs, using script dependent rules and font.

Complexity in Indian scripts:

- Many to many correspondences between character code and font code.
- The character code sequence and the display sequence can differ and hence repositioning may be required at font level.



While handling display in roman script, presence of any character has no effect on the other characters adjacent to it. So, whatever the sequence, the basic characters will be displayed in the same order. Roman script has a one to one correspondence between character code and font code.

What is a Font ?

Collection of glyphs and its associated information such as glyph ID, glyph names, matrix, etc. Usually glyphs that share certain common aesthetic & other

design similarities are part of single font. A font family is a set of fonts that represents some design idea i.e features by which a character's design is recognized. "DV Yogesh" is a font family, even called typeface or simply face. "DV Yogesh Italics " is a font. Fonts are said to belong to one font family, wherein the font(s) share common design but have different attributes such as width, weight, etc. Devanagari Yogesh family of font may have Normal, Italic, Bold & Bold Italic fonts of style Yogesh.

As seen we have character encoding & font encoding. However to render the font into to the display or any other medium requires mapping from character to font code. Since Indian languages are complex in nature, its not one to one mapping of the encoding but also consists of certain language rules as seen above.

Fonts are of various types: 8-bit, 16-bit, etc. Most commonly 8 bit fonts are the True Type fonts. There is a native support of True Type Rasteriser in most of the common operating systems. Open Type fonts are 16 bit fonts, wherein you have lot of flexibility & one can truly represent the language(s). In 8-bit fonts you have only 256 code points available. Out of these first 127 are reserved for English display, out of which first 32 are for reserved by the system. While certain more code points are required by various operating systems especially Unix, Linux, etc. So finally if you really see then out of the 256 code points approx. 96 code points are available for accommodating various glyphs for the given Indian language so as to truly represent the same. Font containing glyphs for English & one given Indian language is called Bi-lingual font, whereas if the code points reserved for English is also being used by Indian language, then the font is termed as mono-lingual font.

Since because of the limitations of the availability of the code points, especially in the 8-bit font, the Font designer needs to do lots of compromises in regards with the representation.

Bitmap fonts & Outline Fonts

Bitmap Fonts (Raster Font)

Bitmap fonts, also called as Raster font are fixed type fonts which are represented by array of dots / pixels, wherein there is no possibility of scaling the fonts to suit the output medium / resolution. Essentially bitmap represent each glyph as a grid of pixels. Bitmap indicates which pixels to make on & off to display a particular character. It is nearly impossible to change the size, shape & resolution of the bitmap character without losing the aesthetics. They are mainly useful for the low-resolution devices & also for the devices wherein there is a limited processing power.

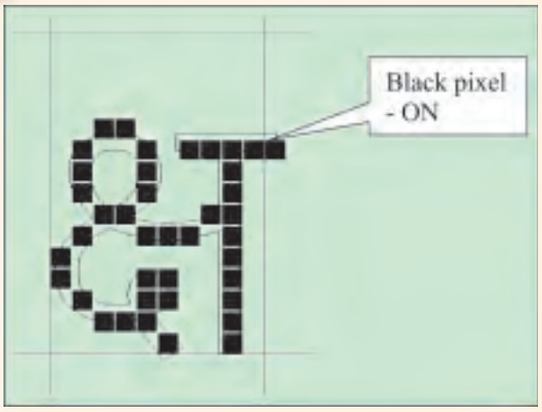
Because of the complex nature of Indian language and because of hinting limitations & failure of rasterisation logic at smaller resolutions, bitmap fonts are preferred choice.

Though there are variety of automatic tools available to convert the outline fonts to bitmap fonts, at lower resolutions, especially for the Indian languages, these tools fails. Though the best option is to hand edit the fonts, for complex scripts lots of skills are required. One pixel on or off can change the readability, shape of the character / glyph. Also it's a cumbersome process, especially when the horizontal & vertical pixel distance is not same.

LaserJet .SFP and .SFL files, TeX PK, PXL, and GF files, Macintosh Screen Fonts, and GEM .GFX files, & .bdf examples of bitmap font formats.

Pixel are dots that represent the smallest units displayed on a computer screen. Typical monitors display about 72 pixel / inch. Characters and graphics are created by turning pixels on / off as shown below.

Figure : Bitmap Glyph for the Devanagari



characters KSha.

Outline Fonts (Vector Fonts)

There are two major font standards in widespread use today the Type 1 & True Type. Now recently Open Type font standards are gaining grounds. In outline fonts each character is represented by mathematical equation which consist of series of lines, curves, etc. When a character from an outline font is to be displayed or printed, the rasteriser engine must on the fly convert the same to bitmap depending upon the resolution of the output medium. The underline hardware or equipment must have built-in rasterizer. The beauty of the technology is that the same font can be adjusted for different resolutions and different aspect ratios.

PostScript Type 1, Type 3, and Type 5 fonts, TrueType fonts, Open Type, Sun F3, MetaFont .mf files, and LaserJet .SFS files are all examples of outline font formats.

Postscript fonts are based on cubic spline technology while True Type is based on quadratic technology. Any quadratic spline can be converted to a cubic spline with no loss. A cubic spline can be converted to quadratic spline with slight loss. It means that it is easy to convert True Type outlines to Postscript outlines, harder to convert Postscript to True Type.

There are generally three font formats used in Adobe PostScript printers: Type 1, Type 3, and Type 5. Type 1 fonts are Adobe's downloadable format. Type 3 fonts are third-party downloadable format. Type 5 fonts are the ROM-based fonts that are part of your printer.

There is no functional difference between a Type 1, Type 3, or Type 5 font. A Type 3 font can do anything a Type 1 or Type 5 font can do.

Type 5 fonts are special in that they often include hand-tuned bitmaps for the commonly used sizes, such as 10- and 12-point. Other sizes are generated from the outlines in normal fashion.

Type 1 fonts, often-called "PostScript fonts," developed by Adobe were initially meant for printing, however through certain utilities they are now available to enable on-screen display.

The TrueType format, originally developed by Apple, was meant for screen & paper medium, and it is supported by varieties of operating systems like Windows, Macintosh, Unix, etc. True Type is an open standard and for having quality fonts displayed on low resolution provide the hinting mechanism.

The hinting mechanism in both the True Type font & Type 1 font is totally different. Type 1 font rasteriser takes advantage of the hints corresponding to particular typographic feature of glyph such as stems, bowls, counters, and so on & produces the best output.. However it contains explicit instructions for shifting control point so as to fit pixel grod. The rasterizer needs to understand these instructions to do so as to exactly turn on pixel(s) for a character at a given size. Apple released TrueType to the world in March 1991 & Microsoft introduced TrueType into Windows with version 3.1 in early 1992.

Scaling PostScript fonts on current versions of the Mac or Windows essentially requires the Adobe Type Manager (ATM) software, which handles the rasterizing to the screen, and rasterizes or converts the fonts for non-PostScript printers. Technically, ATM is not required to use PostScript fonts on PostScript printers, but ATM is required to display the font accurately on screen at arbitrary sizes.

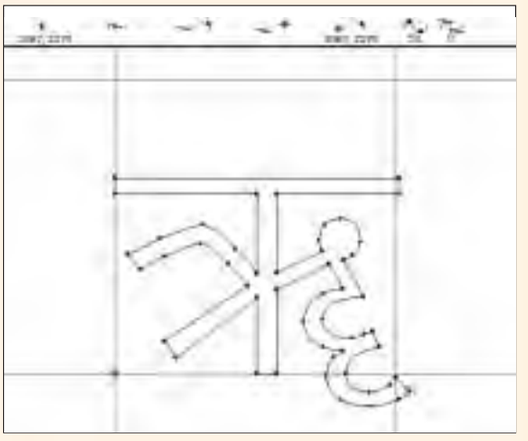


Figure : Outline of Devnagari Glyph Ru.

Open Type

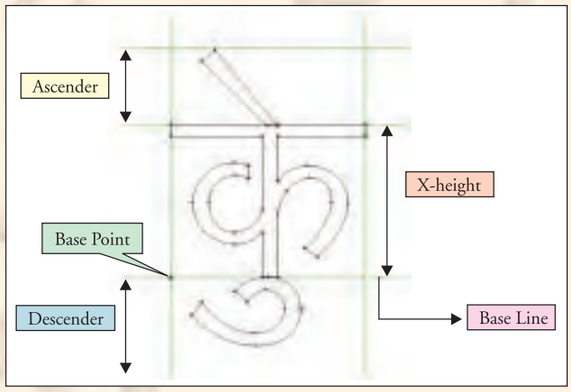
Open Type fonts are outline (scalable) fonts. They are having either PostScript or TrueType outline in a TrueType-style wrapper.

The Open Type format supports most of the advanced features of existing TrueType and PostScript formats, such as Multiple Master fonts (with PostScript outlines), Unicode character sets, and extended character sets to support ligatures, fractions and alternate glyphs. Open Type supports automatic glyph substitution so that one glyph can be substituted for a set (such as the f-f-i ligature, or many Arabic characters), or multiple glyphs can be substituted for a single one. Glyph substitution can happen programmatically or explicitly by user.

Following are the various font file formats

- .afm : Adobe Type 1 metric information in 'ascii' format (human parsable)
- .bdf : Adobe's Bitmap Distribution Format. This format can be converted to the platform specific binary files required by the local X Windows server. This is a bitmap font format distributed in ASCII.
- .bez : Bezier outline information
- .fnt : Bitmapped GEM font in either Motorola or Intel format.
- .fon : An MS-Windows bitmapped font.
- .fot : An MS-Windows kludge for TrueType fonts. The .fot file points to the actual TrueType font (in a ttf file).
- .pfa : Adobe Type 1 Postscript font in ASCII format (PC/Unix) Suitable for directly downloading to your PostScript printer.
- .pfb : Adobe Type 1 PostScript font in "binary`' format (PC/Unix) Not suitable for downloading directly to your PostScript printer. There are utilities for conversion between PFB and PFA
- .pfm : Printer font metric information in Windows format
- .sfl : LaserJet bitmapped softfont, landscape orientation
- .sfp : LaserJet bitmapped softfont, portrait orientation
- .sfs : LaserJet scalable softfont
- .ttf : An MS-Windows TrueType font.
- .ot : Open Type font

Though it is not intended here to teach how the fonts are designed, but would like to given certain standard definitions used in the process of fonts designing. There are variety of tools / utilities available from company's like Macromedia, Fontlab, Adobe, for designing fonts right from the Type 1 to True Type to Open Type for various operating systems. These companies also provides certain tools for easy conversion of font from one format to another.



EM Square :

Each Character fits into a rectangle called em Square. The em units is relative coordinates rather any specific physical distances. Generally speaking for designing True Type fonts em units of minimum 1024 is taken. The font designer can create in that space. The scaled size for rasterized outlines (bitmaps) is referred to in terms of pixels per em (ppem), or the number of pixels occupied by an em-square at that point size. At 72 dpi, the resolution of a Macintosh screen, the ppem and point size are equal.

Glyph:

The actual shape of a character image. For example, Devanagari normal Ka क & Devnagari italic bold Ka **क** are two different glyphs representing the same underlying character ka.

Point : Font size is measure in points. Though the point was set to be 1/12 if a pica and an 83 pica was made equal to 35 centimeters. The measure of one point was 72.27 / inch. For simplicity sake Postscript defined a point as exactly 1/72 of inch.

Baseline

The baseline is an imaginary line upon which each character rests. Characters that appear next to each other are (usually) lined up so that their baselines are on the same level. Some characters extend below the baseline as well, for example the u-matras in Devnagari.

Ascent : A font's maximum distance above the base line is called its ascent.

Descent : A Fonts maximum distance below the base line is called its descent

Bezier curve:

Mathematical equations commonly used to describe the shapes of characters in electronic typography. The Bezier curve was named for Pierre Bezier, a French computer scientist who developed the mathematical representation used to describe the curve.

Monospaced : Like typewritten characters, these all have the same width and take-up the same amount of space. Use of this type allows figures to be set in vertical rows without leaving a ragged appearance (as opposed to proportional spaced)

Proportionately spaced type : Type whose character widths vary according to the features of the letters (as opposed to monospaced type)

ATM:

Adobe Type manager: The program that improves your screen font display by eliminating jagged edges on Type 1 fonts.

Postscript:

Adobe systems page description language. Programs like Macromedia freehand use postscript to create complex pages, text and graphics on –screen. This language is then sent to the printer to produce height quality printed text and graphics.

Rasterization:

The process of converting outlines into bitmaps. The outlines are scaled to the desired size and filled by turning pixels on inside the outline.

Ligatures

A ligature occurs where two or more letterforms are written or printed as a unit. One of the most common ligatures is "fi". Since the dot above a lowercase 'I' interferes with the loop on the lowercase 'F', when 'f' and 'i' are printed next to each other, they are combined into a single figure with the dot absorbed into the 'f'.

Anti-aliasing

On low-resolution bitmap devices (where ragged, ugly characters are the norm) which support more than two colors, it is possible to provide the appearance of higher resolution with anti-aliasing. Anti-aliasing uses shaded pixels around the edges of the bitmap to give the appearance of partial-pixels which improves the apparent resolution.

Character

The smallest component of written language that has semantic value. Character refers to the abstract idea, rather than a specific shape, though in code tables some form of visual representation is essential for the reader's understanding.

Kerning

Kerning refers to kern pairs. It is used to adjust the inter character spacing in certain character groups to improve their appearance. Some letter combinations ("AV" and "To", for example) appear farther apart than others because of the shapes (e.g., A and V) of the individual letters. The typical use of a kern pair is to remove excessive space between a pair of characters

Softfont :

A softfont can be either bitmapped or scalable font, which can be downloaded to your printer. Softfonts uses memory inside the printer. The more the number of fonts downloaded to the

printer, the printer performance for building complex pages gets reduced.

Optical Scaling :

Optical Scaling modifies the relative shape of a character to compensate for the visual effects of changing a character's size. As a character gets smaller, the relative thickness of strokes, the size of serifs, the width of the character, the inter-character spacing, and inter-line spacing should increase. Conversely, as a character gets larger, the relative thickness, widths, and spacing should decrease. Contrast this with linear scaling, in which all parts of a character get larger or smaller at the same rate, making large characters look wide and heavy (strokes are too thick, serifs are too big) while small characters look thin and weak.

Support of Indian languages on the web medium

Text in Indian languages can be displayed on a client machine employing one of the following way :

1. Manual download of font by the user which is provided on the server.
2. .exe file – that takes care of the font installation part. .exe file keeps all the information and data required for font installation. .exe file requires an explicit invocation from user.
3. using activeX control : ocx controls take care of the font installation on the client. An ocx control doesn't require any explicit call from the user for invocation. Invocation is spontaneous as the page gets downloaded and it can't find the required control entry in the registry.
4. Dynamic/Embedded fonts - .eot or .pfr files. in this process no font installation is done. Each time, a page using these files is referenced, font files are downloaded simultaneously along with the page. These files make it possible to display a web page in the font used by designer. The page will be displayed in the required font even if the user doesn't keep that particular font installed on their computer.

Problems with fonts on web medium:

For displaying Indian languages in the client browser, the relevant font needs to be present onto the client machine. This can be achieved by the above mentioned technologies. However, still certain characters are not getting displayed properly, or shows junk, especially when you are using 8 bit True Type fonts. This is because the browser may have utilized the code page which you have allocated the font glyph. Unicode supported Open Type font may solve the issues to certain extent but requires Windows 2000, XP Operating systems having builtin Open Type rasteriser.

(Courtesy : Sh. Mahesh Kulkarni
Coordinator GIST Group,
Centre for Development of Advance Computing
Pune University Campus, Pune)