13. Font & Font Encoding

Introduction

Every language has a set of alphabets through which all words/phrases of that language can be written or represented. Computer is an electronic device, which stores and processes only digits 0 and 1. That s why, a code set (set of decimal code numbers, one for each alphabets), is used for storing, processing and representing text of any language on computer. Ex. - ASCII, ISCII.

To display those codes to its character equivalents on computer s monitor, a visual rendering technique is used. There is one to one mapping for linear scripts. But for non-linear scripts like Indian Languages, where more than one font are available for one code, mapping is not one to one. Thus, for non-linear scripts, a Font Code Set is required for rendering alphabets on computer s screen.

Character Set: -

A character set is a collection of symbols in a specific order, where as the Character is the smallest component of written language, which 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.

A character is a printable symbol having phonetic or pictographic meaning and usually forming part of a word of text, depicting a numeral, or expressing grammatical punctuation.

Example:- ASCII is a character set defines 128 characters. It includes all printable characters from English language as a-z (97-122), A-Z (65-90), 0-9 (48-57), + (43) and & (38) and so on. It also includes 32 non-printable control characters such as Return and Line feed. You generate these characters on the keyboard by holding down the Control key while you strike another key. For example, Bell is value 7, generated by pressing Control plus G (shown as ^G).

Glyph: -

Glyph is the actual shape (bit pattern) of a character image. For example, italic *k* and roman k are two different glyphs representing the same underlying character. Thus, we can say, glyph is a synonym for character image , or simply image . In this broad usage, two images would constitute the same glyph whenever they have essentially the same topology (as in oblique a and roman a), but different glyphs when one is written with a hooked top and the other without (the way one prints an a by hand). In this usage, glyph is a synonym for glyph type .

The glyph for a particular letter of the alphabet may be kept in any location but a table relating the ASCII code and the glyph location may be included in the font file. This gives the freedom for the font designer to place the glyphs in an order that gives some convenience in respect of the design. This way, even if all the glyphs are not present in the font, it will be possible to relate a character to its glyph.

Glyph is a particular graphical representation of a character. Different glyphs can represent the same character. For example, the minuscule letter *a* can be seen in two variants, with a hook at the top, and without.

Glyphs set containing the requisite number of glyphs of vowels, vowel signs, vowel modifiers, consonants, consonant modifiers, numerals, punctuation marks and other required symbols/signs as per requirement of a given script/language.

Type Face:-

Type face is the features by which a character s design is recognized. It contains the digital images of lettering style in 0 s & 1 s. Typeface consists of a co-ordinated set of character designs, usually comprises an alphabet of letters, a series of numerals and a selection of punctuation marks.

Thus, typeface, in its widest sense, is a set of design rules (i.e. a style, look or feel), within which a designer could conceive any character. Typeface is a design for a set of printer or display fonts, each for a set of characters, in a number of specific sizes.

One can sub-divide typefaces of font into two main categories:

1. **serif** (Ex. Times New Roman : This is in Times New Roman)
2. **sans-serif** (Ex. Arial : This is in Arial typeface.)

Typefaces with serifs are often considered easier to read in long passages than those without. The most of this effect is due to the greater familiarity of serif typefaces. As a general rule, printed works such as newspapers and books almost always use serif fonts, at least for the text body. The websites that do specify a font, most use modern sans-serif fonts such as Verdana, because it is commonly believed that, in contrast to the case for printed material, sans-serif fonts are easier than serif fonts to read on computer screens due to their lower resolution.

Serif Vs Sans-Serif

Serif fonts have slabs or strokes on the top and bottom of each character whereas Sans-serif fonts are those with straight edges on the letters.

Font: -

A font is a set of printable or displayable text characters in a specific style and size. The type design for a set of fonts is the typeface and variations of this design form the typeface family.

Example: DV Yogesh is a typeface family, DV Yogesh italic is a typeface, and DV Yogesh italic 10-point is a font.

ASCII, ISCII, UNICODE etc come into picture in back-end for storing and processing of text characters but, for front end visual rendering, font comes into picture.

Thus, a font is a complete set of glyphs including letters, numbers and all special symbols in a particular type and style. The type or typeface is named according to the design such as Arial, or Times New Roman or Courier. The style can either be plain, italic or bold.

Different Types of Fonts:-

1. Bitmap Font: -

Bitmap font stores each glyph as an array of pixels (that is, a bitmap). It is also known as a raster font or fixed type font. Bitmap indicates which pixels to make on-off to display a particular character. Bitmap fonts look best at their native pixel size. At non-native sizes, many text rendering systems perform nearest-neighbor resampling, introducing ugly jagged edges. So, this technique is not well for increasing the size, as it tends to blur the edges.

2. Outline Font:-

Outline font is a set of lines, curves and polygons, as opposed to a bitmap font.

Outline font characters can be scaled to any size easily and better than a bitmap font. It also produce with more attractive results, though this requires a lot of numerical processing.

The result of transforming a character in an outline font in a particular way is often saved as a bitmap in a font cache to avoid repeating the calculations if that character is to be drawn again.

outline fonts are of two type:-

2.1 True Type

2.2 Post Script Type 1

2.3 Open Type Font

2.1 True Type Font: -

TrueType font technology consists of two components:

i. TrueType Fonts and

ii. TrueType Rasterizer.

The TrueType Rasterizer is a piece of software determines the appearance of the letterform in font. Both components are necessary for printing and displaying TrueType fonts. The TrueType format was designed to be efficient in storage and processing. It was also designed to be extendible, making it easy to add improvements and features to the basic technology.

TrueType font files are made up of a number of **tables**, some mandatory (such as the glyf table containing glyph outlines), and some optional. TrueType was designed this way to make the format extensible. TrueType font file contains information such as outlines, hinting instructions, and character mappings (which characters are included in the font). Available for both the Mac and Windows formats, there are slight differences in the TrueType fonts designed for each OS.

True Type Fonts are mostly 8 bit fonts, so only 256 code points are available. Out of 1st 127 are reserved for English display and some code point for special requirement of OS. Thus, approximately, 96 code points are available for accommodating various glyphs for given Indian language. Due to limitation of availability of code points, font designer needs to do lots of compromises in regards with the representation.

2.2 PostScript Type1 Font: -

PostScript Type1 Font is developed for post script printer, by adobe.

There are two parts to a PostScript font: (i.) bitmapped screen information that displays the font on a computer monitor, and (ii.) information that tells a PostScript printer how to reproduce the font on paper, or whatever medium thats in your printer. Like TrueType fonts, PostScript fonts are scalable (the characters can be enlarged or reduced), but PostScript fonts are only scaled automatically when output to PostScript printers. For scaling fonts on-screen or for outputting to non-PostScript printers, Adobe Type Manager (ATM) must be installed on your computer, it is offered free of charge.

2.3 Open Type Font:-

Open Type font is super set of True Type Font and PostScript Type1 Font format. Open Type fonts can include an expanded character set and layout features, providing broader linguistic support and more precise typographic control. Open Type fonts can be installed and used alongside PostScript Type 1 and TrueType fonts.

The three main benefits of the Open Type format are

- Its *cross-platform compatibility* (the same font file works on Macintosh and Windows computers)
- Its ability to support widely expanded character sets and layout features, which provide richer linguistic support and advanced typographic control.
- Better Font Protection

  Open Type fonts also contain a digital signature that allows operating systems and browsing applications to identify the source and integrity of fonts including embedded font files obtained in Web documents.

  Open Type provides a series of enhancements to the TrueType format, the most significant of which allows PostScript font data to nest inside a TrueType software wrapper.

TrueType fonts held the most advantages for on screen use, while high resolution imaging has been the domain of Type 1 fonts. Where as Open Type can give you the best of both technologies, such as Multiple Master Fonts, Unicode character sets and extended character set to support ligatures, fractions and alternate glyphs.

Open Type is a font with data describing how the glyphs are to be selected and rendering on the screen or display device. The Open Type font data is in the form of tables of rules (to reorder characters, map from characters to glyphs, make contextual substitutions & position marks and other glyphs in relation to each other) inside the font file which required a software program to read the tables for rendering process. This is the distinctive part of this font to other fonts, having a direct correspondence between the data characters and the displayed glyphs, where glyphs are laid out simply in a straight line, side by side.

Open Type fonts have property of KERNING, which is the intelligent spacing of certain pairs of letters. Kerning improve the result as

निमाज्जुद्दीन (without kerning)

निमाज्जुद्दीन (after kerning)

Open Type font technology has a complex glyph processing model, which converts **original string of glyphs** into a **new string of glyphs**. In smart glyph processing, the glyph stream passes through three key phases:-

i. **Substitution** ensures that the glyph string consists of the right glyphs in the right order in the string.

ii. **Positioning** ensures that the glyphs are correctly positioned.

iii. **Justifications** (optional) - increase or decrease the space taken up by the string when it is rendered

3. Dynamic Font: -

Dynamic fonts are font style files that download right along with the page that will use them. Dynamic font file will download into your cache just like an image. Once there, as long as you dont clear your cache completely, the font will be there for all future visits. It is fun to watch the page the first time you use these fonts. The page will come in fully with the text in the default format. Then, once the font, or fonts, gets downloaded, the entire page reloads and comes to life.

4. Intelligent Font: -

Intelligent fonts alter their appearance depending on their linguistic context.

Computerized Typography allows the introduction of aeolotropic fonts, which change their physical properties when their position changes. Thus applying phonetic or grammatic rules may control a character s appearance. The context dependence of aeolotropic fonts distinguishes them from random fonts that have an aleatoric or random appearance.

Font Design:-

The Indian script uses a vast variety of glyph. The application of vowel modifiers (matra) changes glyph according to the consonant to which it is applied. Similarly some consonants each other and form a totally different glyph. There are many features required in the font design to represent the script correctly and adequately. Some of these are presented here-

Issues in Font Design: - here are few important points that may be taken under consideration in font design-

1. There is a requirement of displaying, encoding (into character set) and glyphs set for not only *akshar* but also for symbols of punctuation and *matras* (of different widths due to variable width of *aksharas*). So, there is a need to cover all and every possible shapes in the font set.

2. If vowels and consonants are coded in eight bits, most of the application program can deal with these fonts, but now, one to one mapping between an *akshara* and a glyph is ruled out due to availability of only 256 code space (point).

3. A minimum of 300 glyphs are required to display all samyuktaksharas (consonant conjuncts).

4. A complex conjunct may have up to four consonants and a vowel. It can be replaced as a unique shape with just one specially designed glyph. Now, this conjunct will contain multiple bytes. Thus we still have to deal with the problem of multiple byte representations for the aksharas in our language.

5. There are more than 13000 different codes for our languages; thus, One to One mapping from akshara to glyph requires sixteen bit codes for the aksharas. There is also a need of fonts developed to accommodate that many glyphs. Solution of this problem requires a computer system can deal with 16 bit character code, which index into 16 bit font and can display the glyph.

6. The ASCII strings required to display Devanagari text differ for each font. The number of glyphs required to display an akshara depend on the akshara (or samyuktakshara) and thus a variable number of bytes is required to represent each akshara. This is in contrast with the western alphabet where one letter is displayed through just one glyph.

The following five steps were followed in designing the Saral series at C-DAC, Mumbai under the type font design directorship of Prof. R. K. Joshi, a well known personality in field of digital type design and text processing issues in Indian languages-

Step1- Planning the lettering style of the type font. Identifying and designing outlines of the glyphs as per the character codes of Unicode.

Step2 - Identifying and designing outlines of the total glyph set as per the linguistic and typographic requirement of a language/ script.

Step3 - Writing adequate number of composition rules and tables to support the correct construction of syllables and text.

Step4 - Integrating the tables and the font into the operating system/application.

Step5 - Testing of the glyphs and linguistic syllables using the font, tables, input and output devices and O/S.

Font Encoding: -

The mapping between the characters codes (an integer value) to the characters are called as font encoding.

Example:- Either a character with code 92 wind up printing as a backslash (as it does under the ASCII encoding) or as a double left quote (as it does under the most common TeX font encoding). So in another way, the font encoding is the arrangement of the characters in the font.

The concept of font encoding allows us to generate displays of text strings in many different languages by designing fonts, which contain the glyphs corresponding to their alphabet. The text to be displayed is represented as a series of eight bit characters and for all practical purposes, these may be reckoned as a string of ASCII codes. The computer system takes each code and displays the glyph associated with it. The glyphs may be viewed as the building blocks for the letter to be displayed where, by placing the glyphs one after another, the required display is generated. Fonts also incorporate a feature whereby some of the glyphs may be defined to have zero width even though they extend over a horizontal range. Thus when the system places a zero width glyph next to another, the two are superimposed and thus permit more complex shapes to be generated, such as accented letters. Zero width glyphs are very important for Indian language fonts.

Finally, Font encoding is not same as internal storage representation of a character. Especially, for non-linear Indic Scripts, where vowel ending of a consonant or consonant-cluster transforms its shape as matra. For example:

1.	**2.**
क + इ = कि	स्थिति = स्थि + ति
C V = CM	= (स् थ् इ) (त् इ)
k i = ki	(CCV) (CV)
	(CCM) (CM)
	sthi ti

In the first example:

क = क् + इ

Linguistically, it is ok to show that Hindi akshar कि (ki) is made by combining क् (k) and Independent vowel इ (I). Unicode value of क् (k) is (U+0915 + U+094D) as M (k) is combination of क (ka) (U+0915) and sign ् (U+094D) where as Unicode value of vowel इ (I) is U+0907.

But actually for storage purpose, akshar कि will be a combination of akshar क (ka) and dependant vowel sign ि. Its respective Unicode values are U+0915 and U+O93F.

In the second example:

We can observe that the shape of matra ि (I) is not same in two Hindi words कि and स्थिति. The shape of matra ि is wider in akshar स्थिति than akshar कि while the Unicode value of matra ि is (U+093F) same in both akshar. This is an important property of script like Indian Languages.

ISFOC

C-DAC developed the character-slice (glyph) coding standards to ensure good appearance and aesthetics for which Indian scripts are famous. Unlike ISCII, these code charts are different for each script and are represented in 8-bits only. They are suitable for all GUI environments requiring bit-map fonts, Type-1 fonts or True Type fonts.

Intelligence Based Script Font Code ISFOC is a coded character set containing all the basic shapes required for rendering a script. These shapes can be overlapped linearly to compose any word in the script. Each ISFOC set can contain a maximum of 188 characters. This is adequate for most of the scripts. However, some require more. ISFOC provides the code set for inclusion of complex scripts in graphics-oriented environments like MS-Windows and Macintosh. The basic shapes are chosen such that they can represent a wide variety of font styles in the script.

Since ISFOC fonts are linearly composed, they can be used along with the existing English applications and printed on existing Laser printers and Typesetters.

ISFOC is an 8 bits font encoding for rendering all possible shapes of a particular script on computer screen. It is a C-DAC s recommendation for rendering any script. It is not a standard and mostly used in CDAC products. It doesn t support 16 or 32 bits character codes, so it is not much relevant after popularity of Unicode -16 and 32.

INSFOC

INSFOC stands for Indian Standard Font Code. The Font is laid out such that the font remains unchanged between the character locations 0x80(127) to 0xFF(255) in monolingual and bilingual font layout.

Font containing glyph for English and one Indian language is called Bilingual Font but if the code point reserved for English is also being used by Indian language, then the font is called as Mono- Lingual font.

INSFOC font standards are targeted towards the following class of users as Data Processing, Office Users / Word Processing, Textbook Publishers, Web Content Creators, Desktop Applications. It is certainly not targeted towards professional desktop publishers, advertising agencies and highly Sanskritized text content creators. INSFOC allows opening of text composed in some font of an editor to be opened with another font in another editor.

References:-

- http://en.wikipedia.org
- http://cdac.in
- http://tdil.mit.gov.in
- http://acharya.iitm.ac.in

Courtesty :
Manoj Jain & Vijay Kumar
mjain@mit.gov.in

Department of Information Technology
New Delhi-110003

Ashutosh Chaurasia
CDAC, Noida