

---

# A Guide To Indian Language Font Development

Elaborate, Illustrated font development guideline document for scripts of  
India by font designers, developers, language experts.

---

*Editor*

Santhosh Thottingal

*Contributors*

Balasankar C

Behdad Esfahbod

Hrishikesh

Kartik Mistry

Kavya Manohar

Rahimanuddin Shaik

Rajeesh Nambiar

Ryan Kaldari

Vasudev Kamath

March 31, 2014

<https://github.com/IndicFontbook/Fontbook>



Dedicated to all font designers



---

# Contents

---

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Objectives of document . . . . .	2
1.2	Target Audience for this document . . . . .	2
1.3	Scope . . . . .	2
1.4	How to use this document . . . . .	3
1.5	Notes on Collaboration . . . . .	3
<b>2</b>	<b>General Concepts</b>	<b>5</b>
2.1	Script . . . . .	5
2.2	Complex script . . . . .	5
2.3	Glyph . . . . .	6
2.4	Ligatures . . . . .	7
2.5	Cluster/Syllable . . . . .	8
2.6	Akhand . . . . .	8
2.7	Ra Forms . . . . .	8
2.8	Split Matra . . . . .	8
2.9	Reordering . . . . .	9
2.10	Zero Width Joiner . . . . .	9
2.11	Zero Width Non Joiner . . . . .	9
2.12	Stacking . . . . .	9
<b>3</b>	<b>Open font format</b>	<b>11</b>
3.1	Introduction . . . . .	11
3.2	GPOS . . . . .	11
3.3	GSUB . . . . .	11
3.4	GDEF . . . . .	11
3.5	Shaping Engine . . . . .	11
3.6	Shape Glyph sequence . . . . .	12
3.7	Position Glyph sequence . . . . .	12

3.8	Reference fonts . . . . .	12
3.9	Reference Rendering Engine . . . . .	12
<b>4</b>	<b>Bengali</b>	<b>13</b>
4.1	Introduction . . . . .	13
4.2	Reference fonts . . . . .	13
4.2.1	Lohit Bengali . . . . .	13
4.2.2	History . . . . .	13
<b>5</b>	<b>Devanagari</b>	<b>15</b>
5.1	Introduction . . . . .	15
5.2	Reference fonts . . . . .	15
5.2.1	Lohit Devanagari . . . . .	15
5.2.2	History . . . . .	16
<b>6</b>	<b>Gujarati</b>	<b>17</b>
6.1	Introduction . . . . .	17
6.2	Reference fonts . . . . .	17
6.2.1	Lohit Gujarati . . . . .	17
6.2.2	History . . . . .	18
<b>7</b>	<b>Kannada</b>	<b>19</b>
<b>8</b>	<b>Malayalam</b>	<b>21</b>
8.1	Introduction . . . . .	21
8.2	Orthography variation . . . . .	22
8.3	Reference fonts . . . . .	23
8.3.1	Meera Font . . . . .	23
8.3.2	Lohit Malayalam Font . . . . .	24
8.3.3	History . . . . .	24
8.4	Technical details . . . . .	25
8.4.1	Opentype Script Tags - mlym and mlm2 . . . . .	25
8.4.2	Reordering . . . . .	25
8.4.3	Vowel signs and combining marks . . . . .	25
8.4.4	Samvruthokaram . . . . .	28
8.4.5	Conjunct Signs for , , . . . . .	30
8.4.6	Reph . . . . .	30
8.4.7	Dot Reph . . . . .	31
8.4.8	Chillus . . . . .	32
8.4.9	Stacking . . . . .	33
8.4.10	Font metrics . . . . .	34
8.4.11	Positioning rules . . . . .	34

8.4.12	ZWNJ and ZWJ Signs . . . . .	34
8.4.13	Prebase substitutions . . . . .	34
8.4.14	Akhand forms . . . . .	34
8.4.15	Below base forms . . . . .	34
8.4.16	Below base substitutions . . . . .	34
8.4.17	Half forms . . . . .	34
8.4.18	Postbase forms . . . . .	34
8.4.19	Latin glyphs and punctuations . . . . .	34
8.4.20	Kerning . . . . .	34
8.4.21	Shape references . . . . .	34
8.4.22	Left and right bearings . . . . .	34
8.4.23	Italic variant . . . . .	34
8.4.24	Bold variant . . . . .	34
8.5	Design . . . . .	34
8.5.1	Number of glyphs . . . . .	34
8.5.2	Guidelines . . . . .	35
<b>9</b>	<b>Odiya</b>	<b>39</b>
<b>10</b>	<b>Panjabi</b>	<b>41</b>
<b>11</b>	<b>Tamil</b>	<b>43</b>
<b>12</b>	<b>Telugu</b>	<b>45</b>
12.1	Introduction . . . . .	45
<b>13</b>	<b>Glossary</b>	<b>47</b>
<b>14</b>	<b>References</b>	<b>51</b>
<b>15</b>	<b>Appendices</b>	<b>53</b>





---

## Introduction

---

"I can't go to a restaurant and order food because I keep looking at the fonts on the menu.-Donald Knuth"

---

One of the integral building blocks for providing multilingual support for digital content are fonts. In current times, OpenType fonts are the choice. With the increasing need for supporting languages beyond the Latin script, the TrueType font specification was extended to include elements for the more elaborate writing systems that exist. This effort was jointly undertaken in the 1990s by Microsoft and Adobe. The outcome of this effort was the OpenType Specification - a successor to the TrueType font specification.

Fonts for Indic languages had traditionally been created for the printing industry. The TrueType specification provided the baseline for the digital fonts that were largely used in desktop publishing. These fonts however suffered from inconsistencies arising from technical shortcomings like non-uniform character codes. These shortcomings made the fonts highly unreliable for digital content and their use across platforms. The problems with character codes were largely alleviated with the gradual standardization through modification and adoption of Unicode character codes. The OpenType Specification additionally extended the styling and behavior for the typography.

The availability of the specification eased the process of creating Indic language fonts with consistent typographic behavior as per the script's requirement. However, disconnects between the styling and techni-

cal implementation hampered the font creation process. Several well-stylized fonts were upgraded to the new specification through complicated adjustments, which at times compromised on their aesthetic quality. On the other hand, the technical adoption of the specification details was a comparatively new know-how for the font designers. To strike a balance, an initiative was undertaken by the a group of font developers and designers to document the knowledge acquired from the hands own experience for the benefit of upcoming developers and designers in this field.

The outcome of the project will be an elaborate, illustrated guideline for font designers. A chapter will be dedicated to each of the Indic scripts - Bengali, Devanagari, Gujarati, Kannada, Malayalam, Odia, Punjabi, Tamil and Telugu. The guidelines will outline the technical representation of the canonical aspects of these complex scripts. This is especially important when designing for complex scripts where the shape or positioning of a character depends on its relation to other characters.

This project is open for participation and contributors can commit directly on the project repository.

## 1.1 Objectives of document

## 1.2 Target Audience for this document

## 1.3 Scope

- This document covers only languages and scripts recognized in India.
- This document covers all complex OFF features required for Indian scripts.
- This document is based on the current expertise of community members working in this area.
- This document is from the typography perspective for each script and many not be linguistically correct.
- This document does not cover design or calligraphy style aspects but covers only technical aspects.

- This document is based on Unicode 6.2 and ISO/IEC 14496-22:2009 (Second Edition) "Open Font Format" standard.
- This document is not a tutorial on font design or development and does not teach typography.

## 1.4 How to use this document

Elaborate, Illustrated font design guideline document for Indic fonts by font designers, developers, language experts. - designer freedom to adapt

## 1.5 Notes on Collaboration



---

## General Concepts

---

### 2.1 Script

At least one set of defined base elements or symbols, is a requirement for all writing systems. these symbols are individually termed as characters and collectively called a script. simply the set of the symbols required to represent a writing system is called a script. a script may in turn be used to represent more than one languages. Latin, Devanagari and Arabic are examples of scripts. English, French, German, and Latin are all languages written using the Latin script.

### 2.2 Complex script

Complex script is a writing system in which the shape or positioning of a character depends on its relation to other characters.

What makes it complex? \* Bi-directional: text is written right-to-left (For example: Arabic, Hebrew) and left-to-right (For example: Devanagari)

\*Context sensitive shaping and ligatures: Character may change shape depending upon position.



The Devanagari ddhrya-ligature, as displayed in the  
JanaSanskritSans font.

\* Ordering: In Gujarati, Ki is where 'i' is place before 'K'.

//FIXME What is a complex script? What makes it complex, some examples, screenshots 2-3 paragraph + images How it differs from simple scripts like Latin Character

## 2.3 Glyph

A glyph is an element of writing. It can be a single character or a group of characters. Visually, if you see one or more characters form a single visual unit, it is called a glyph.

In typography, it is "the specific shape, design, or representation of a character".<sup>1</sup>

It is a particular graphical representation, in a particular typeface, of an element of written language, which could be a grapheme, or part of a grapheme, or sometimes several graphemes in combination.

To illustrate this concept, a set glyphs inside a Latin font (Fig. 2.1) and a Malayalam font (Fig. 2.2) as seen in fontforge is given below.

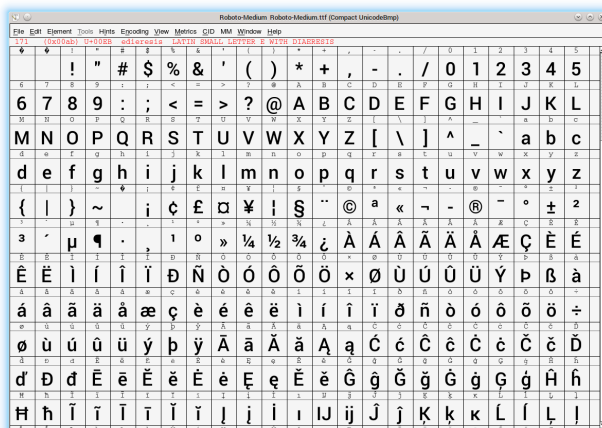


Figure 2.1: Glyphs inside Roboto font

<sup>1</sup>Ilene Strizver. "Confusing (and Frequently Misused) Type Terminology, Part 1". fonts.com. Monotype Imaging.



Figure 2.2: Glyphs inside Meera font

## 2.4 Ligatures

In writing and typography, a ligature occurs where two or more graphemes or letters are joined as a single glyph. Ligatures usually replace consecutive characters sharing common components and are part of a more general class of glyphs called "contextual forms", where the specific shape of a letter depends on context such as surrounding letters or proximity to the end of a line.<sup>2</sup>

By way of example, the common ampersand '&' represents the Latin conjunctive word *et*, for which the English equivalent is the word "and". The ampersand's symbol is a ligature, joining the old hand-written Latin letters *e* and *t* of the word *et*, so that the word is represented as a single glyph.

The Brahmic abugidas make frequent use of ligatures in consonant clusters. The number of ligatures employed may be language-dependent; thus many more ligatures are conventionally used in Devanagari when writing Sanskrit than when writing Hindi. Having 37 consonants in total, the total number of ligatures that can be formed in Devanagari using only two letters is 1369, though few fonts are able to render all of them.

<sup>2</sup>[https://en.wikipedia.org/wiki/Typographic\\_ligature](https://en.wikipedia.org/wiki/Typographic_ligature)

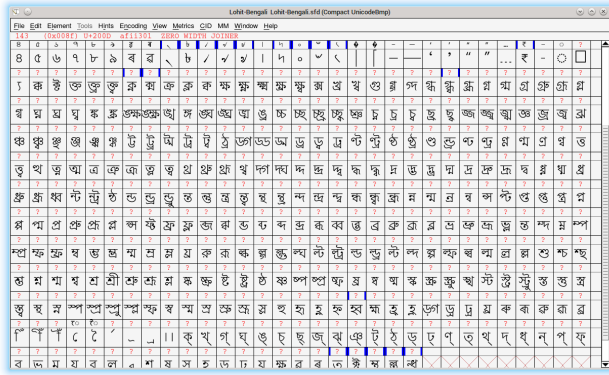


Figure 2.3: Glyphs inside Lohit Bengali font

द्ध्रय

Figure 2.4: The Devanagari ddhrya-ligature (द् + ध् + र् + य = द्ध्रय)

## 2.5 Cluster/Syllable

## 2.6 Akhand

## 2.7 Ra Forms

//FIXME Explain Reph Explain Rakar

## Matra

//FIXME Explain prebase, postbase matra with examples and images

## 2.8 Split Matra



കു

Figure 2.5: The Malayalam kthu-ligature (ക + ൃ + ത + ു )

સ + ં = સં

Figure 2.6: The Gujarati 's' formed from SA and ANUSVARA

## 2.9 Reordering

## 2.10 Zero Width Joiner

## 2.11 Zero Width Non Joiner

## 2.12 Stacking



---

## Open font format

---

### 3.1 Introduction

### 3.2 GPOS

GPOS or Glyph POSitioning table.

[See <http://fontforge.org/gposgsub.html#opentype>]

### 3.3 GSUB

GUSB or Glyph SUBstitution table.

[See <http://fontforge.org/gposgsub.html#opentype>]

### 3.4 GDEF

### 3.5 Shaping Engine

[Some content from <http://behdad.org/text/> can be used ]

### 3.6 Shape Glyph sequence

### 3.7 Position Glyph sequence

### 3.8 Reference fonts

To illustrate the concepts in this document, we will be using a set of reference fonts for each script, some times more than one per script.

List of fonts table

### 3.9 Reference Rendering Engine

We are using Harfbuzz as a reference rendering engine. Examples given are working perfectly with Harfbuzz but any rendering engine conforming to the Open font specification will give same result.

---

# Bengali

---

## 4.1 Introduction

## 4.2 Reference fonts

### 4.2.1 Lohit Bengali

Lohit Bengali is one of the popular fonts for Bengali.

### 4.2.2 History

In 2004, Red Hat released five fonts for the Indian language under GPL 2. The fonts were originally developed by Modular Infotech <sup>1</sup>. In 2011, Red Hat relicensed fonts under SIL OFL 1.1 license <sup>2</sup>. The fonts named Lohit which means Red in Sanskrit. Currently, the font family supports 21 Indian languages: Assamese, Bengali, Devanagari (Hindi, Kashmiri, Konkani, Maithili, Marathi, Nepali, Sindhi, Santali, Bodo, Dogri), Gujarati, Kannada, Malayalam, Manipuri, Odiya, Punjabi, Tamil, and Telugu.

Now, Fedora Project and its contributors took the responsibility to consolidate the further efforts and improvements of the Lohit fonts.

---

<sup>1</sup>Modular Infotech <http://www.modular-infotech.com/>

<sup>2</sup>License change announcement of Lohit fonts <https://www.redhat.com/archives/lohit-devel-list/2011-September/msg00000.html>

Homepage: <https://fedorahosted.org/lohit/>

---

# Devanagari

---

## 5.1 Introduction

Devanagari is an abugida alphabet for India and Nepal. Devanagari is used to write Standard Hindi, Marathi, Nepali along with Awadhi, Bodo, Bhojpuri, Gujarati, Pahari, (Garhwali and Kumaoni), Konkani, Magahi, Maithili, Marwari, Bhili, Newar, Santhali, Tharu, and sometimes Sindhi, Dogri, Sherpa, Kashmiri and Punjabi. It was formerly used to write Gujarati. Because it is the standardized script for the Hindi language, Devanagari is one of the most used and adopted writing systems in the world.

## 5.2 Reference fonts

### 5.2.1 Lohit Devanagari

Lohit Devanagari font is considered as most popular Devanagari font in India. Gargi, Chandas, Kalimati, Samanata, Nakula are other popular free and open-source fonts available.

We will take Lohit Devanagari as reference font.

## 5.2.2 History

In 2004, Red Hat released five Indian language fonts as open source licensed under the GPL. In 2011, Red Hat relicensed fonts under SIL OFL 1.1 license. The fonts named Lohit which means Red in Sanskrit. Currently, the font family supports 21 Indian languages: Assamese, Bengali, Devanagari (Hindi, Kashmiri, Konkani, Maithili, Marathi, Nepali, Sindhi, Santhali, Bodo, Dogri), Gujarati, Kannada, Malayalam, Manipuri, Odiya, Punjabi, Tamil, and Telugu.

Now, Fedora Project and its contributors took the responsibility to consolidate the further efforts and improvements of the Lohit fonts.

Homepage: <https://fedorahosted.org/lohit/>



---

## Gujarati

---

### 6.1 Introduction

Gujarati script is an abugida like many other Indic scripts rather than alphabet. It is used for languages like Gujarati and Kutchi. Gujarati script is variant of Devanagari script differentiated by the loss of the characteristic horizontal line running above the letters and by a small number of modifications in the remaining characters.

The modern Gujarati alphabet has 13 vowel letters, 36 consonant letters, 12 vowel extensions and few other symbols.

### 6.2 Reference fonts

We will use Lohit Gujarati as reference font as of now. Other popular Unicode Gujarati fonts are: Shruti (non free, Microsoft), Rekha, Aakar and Kalapi. Arial MS Unicode is often used as default Gujarati font and Gujarati MT is popular default Gujarati font in Mac OS X.

#### 6.2.1 Lohit Gujarati

// FIXME: Short introduction, designers, maintainers, usage info, popularity of the font.

Lohit Gujarati font is consider as one of the most popular OpenSource Gujarati font.

### 6.2.2 History

In 2004, Red Hat released five Indian language fonts as open source licensed under the GPL. In 2011, Red Hat relicensed fonts under SIL OFL 1.1 license. The fonts named Lohit which means Red in Sanskrit. Currently, the font family supports 21 Indian languages: Assamese, Bengali, Devanagari (Hindi, Kashmiri, Konkani, Maithili, Marathi, Nepali, Sindhi, Santali, Bodo, Dogri), Gujarati, Kannada, Malayalam, Manipuri, Odiya, Punjabi, Tamil, and Telugu.

Now, Fedora Project and its contributors took the responsibility to consolidate the further efforts and improvements of the Lohit fonts.

Homepage: <https://fedorahosted.org/lohit/>

7

---

Kannada

---



---

# Malayalam

---

## 8.1 Introduction

Like many other Indic scripts, it is an abugida, or a writing system that is partially “alphabetic” and partially syllable-based. The modern Malayalam alphabet has 13 vowel letters, 36 consonant letters, and a few other symbols. The Malayalam script is a Vattezhuttu script, which had been extended with Grantha script symbols to represent Indo-Aryan loanwords. The script is also used to write several minority languages such as Paniya, Betta Kurumba, and Ravula. The Malayalam language itself was historically written in several different scripts.

As is the case for many other Brahmi-derived scripts in the Unicode Standard, Malayalam uses a virama character to form consonant conjuncts. The virama sign itself is known as candrakala(ചന്ദ്രകല) in Malayalam.

When the candrakala sign is visibly shown in Malayalam, it usually indicates the suppression of the inherent vowel, but it sometimes indicates instead a reduced schwa sound, often called “half-u” or samvruthokaram. In the latter case, there can also be a -u vowel sign, and the base character can be a vowel letter. In all cases, the candrakala sign is represented by the character U+0D4D Malayalam sign virama, which follows any vowel sign that may be present and precedes any anusvara that may be present.

FIXME samvruthokaram needs more explanation, refer [http://thottingal.in/documents/Malayalam\\_Unicode\\_Report\\_of\\_Workshop\\_Kerala\\_University.pdf](http://thottingal.in/documents/Malayalam_Unicode_Report_of_Workshop_Kerala_University.pdf) and <http://smc.org.in/doc/rachana-malayalam-collation.pdf>

## 8.2 Orthography variation

Malayalam has two orthography variations, both actively used. Generally they are known as Old orthography and Modern orthography. Old orthography is also known as traditional orthography. Modern orthography is also known as reformed orthography.

Old orthography is generally identified by large number of ligature glyphs, often formed by more than one consonants or vowel combination. For example, in old orthography, consonant ക + vowel ു will form കൂ as a single ligature. similarly a conjunct ക + ് + ത will form കത as a single ligature. Font developers of traditional style fonts design and draw this large set of glyphs. For a reference font Meera, the number of glyphs is approximately over a thousand.

In the 1970's and 1980's, Malayalam underwent orthographic reform due to printing difficulties. The treatment of the combining vowel signs u and uu was simplified at this time. These vowel signs had previously been represented using special cluster graphemes where the vowel signs were fused beneath their consonants, but in the reformed orthography they are represented by spacing characters following their consonants.

മലയപ്പുലയനാ മാടത്തിൻമുറ്റത്തു  
മഴ വന്ന നാളൊരു വാഴ നട്ടു.  
മനതാരിലാശകൾപോലതിലോരോരോ  
മരതകക്കുന്യു പൊടിച്ചുവന്നു.  
അരുമകിടാങ്ങളിലൊന്നായതിനേയ്യ -  
മഴകിപ്പുലക്കള്ളിയോമനിച്ചു.

Figure 8.1: Text rendering using traditional orthography. Font used is Meera

The above examples give high level difference in orthography. Since both orthography is prominent in day to day usage of Malayalam, it will be helpful to get more understanding about the orthography variation. So let us go some more deep into this differences.

മലയപ്പുലയനാ മാടത്തിൻമുറ്റത്തു  
മഴ വന്ന നാളൊരു വാഴ നട്ടു  
മനതാരിലാശകൾപോലതിലോരോരോ  
മരതകക്കുമ്പു പൊടിച്ചുവന്നു  
അരുമക്കിടാങ്ങളിലൊന്നായതിനേയും  
മഴകിപ്പുലക്കളളിയോമനിച്ചു

Figure 8.2: Text rendering using modern orthography. Font used is Lohit Malayalam

For vowel signs, only u vowel(ു) and uu vowel(ൂ) make the difference. ു

As you can see, in traditional orthography the vowel sign attach to the consonant, while in modern script, the vowel sign is separate.

Reph sign varies in both orthographies. Reph sign is formed by VIRAMA+ RA i.e, ി് + ര.

As you can see, in traditional orthography the reph sign attach to the consonant, while in modern script, reph sign is separate and it get re-ordered to the left of consonant.

## 8.3 Reference fonts

Since we need to illustrate both orthographies we will be using 2 fonts. For old orthography, we will use Meera font and for modern orthography we will use Lohit Malayalam.

### 8.3.1 Meera Font

മീര മാതൃക // FIXME: Short introduction, designers, maintainers, usage info, popularity of the font.

Meera font is maintained by Swathanthra Malayalam Computing initiative.

Homepage: <https://savannah.nongnu.org/projects/smc>

Traditional:

ക + ു = കൂ

ക + ൂ = കൂ

Modern:

ക + ു = കൂ

ക + ൂ = കൂ

Figure 8.3: Rendering difference of u and uu vowel signs

### 8.3.2 Lohit Malayalam Font

// FIXME: Short introduction, designers, maintainers, usage info, popularity of the font.

### 8.3.3 History

In 2004, Red Hat released five Indian language fonts as open source licensed under the GPL. In 2011, Red Hat re-licensed fonts under SIL OFL 1.1 license. The fonts named Lohit which means Red in Sanskrit. Currently, the font family supports 21 Indian languages: Assamese, Bengali, Devanagari (Hindi, Kashmiri, Konkani, Maithili, Marathi,



Traditional:



Modern:



Figure 8.4: Rendering difference of REPH sign

Nepali, Sindhi, Santali, Bodo, Dogri), Gujarati, Kannada, Malayalam, Manipuri, Odiya, Punjabi, Tamil, and Telugu.

Now, Fedora Project and its contributors took the responsibility to consolidate the further efforts and improvements of the Lohit fonts.

Homepage: <https://fedorahosted.org/lohit/>

## 8.4 Technical details

### 8.4.1 OpenType Script Tags - mlym and mlm2

### 8.4.2 Reordering

### 8.4.3 Vowel signs and combining marks

For every vowels except  $\text{ഌ}$ , there are vowel signs in Malayalam

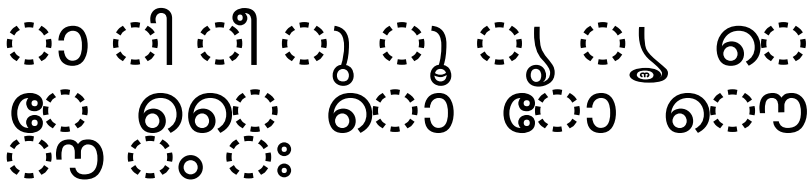


Figure 8.5: Malayalam vowel signs.

The vowel signs cannot stand alone. It always applies to a consonant. To denote that linguistic property, a dotted rectangle appears if we try to render the vowel signs alone.

$$വ + ീ = വി$$

Figure 8.6: Malayalam vowel ീ getting applied.

In the above shown vowel signs ീ, ു, ൂ, ൃ, ൄ, and ൅ needs to be reordered before rendering. Rendering engines will reorder the glyphs before rendering and rendering engine is aware of this property of the above vowels. Inside the font, nothing special need to be done to get this reordering. Fonts just need the glyph.

$$വ + െ = വെ$$

Figure 8.7: Malayalam vowel െ getting applied after reordering.

vowel signs െ, േ and ൈ needs to be spit before rendering. First part will be applied at the beginning of consonant cluster and right part will be applied to end of consonant cluster. In rendering engine terminology these vowel signs are known as "Split Matra".

$$\begin{aligned} വ + െ = വ + െ + െ \\ = െ + വ + െ \\ = വെ \end{aligned}$$

Figure 8.8: Malayalam vowel െ getting spit and applied at left and right of വ .

In the above example, or in the rendering of split matra, the glyph mapped to vowel signs െ, േ and ൈ has no role. Whatever glyph present in the font for vowel signs െ, േ and ൈ will not be used. Instead of that, glyphs mapped to the components after split will be used. So what matters is what glyphs are mapped to vowel signs െ, േ, ൈ matters.

Because of this, fonts will have glyphs to illustrate this process. For example, Meera, Rachana fonts has the following drawing:

We mentioned that dotted circles will appear left of the vowel sign if it is not attached to consonant or consonant cluster. But there are two

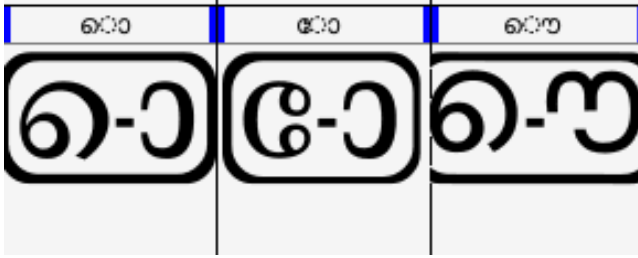


Figure 8.9: Split matra glyphs inside Meera font for ഐ, ഘോ and ഘോ

known exceptions to this rule in Malayalam. One is samvruthokaram. In samvruthokaram - ി virama is applied to a vowel sign and this can be used to cause a rendering as shown in Figure 8.10.



Figure 8.10: Wrong rendering of Samvruthokaram.

This bug was fixed in 2008 and See section 8.4.4 on Samvruthokaram for more details. It now renders properly as seen in Figure 8.11.



Figure 8.11: Correct rendering of Samvruthokaram.

Another exception is ഞ. This combination of a long vowel sign and anusvara is used to denote "nth" like, 16ഞ or 16-ഞ meaning 16th. ഞ sign cannot appear in this example without a dotted circle. To get the expected rendering, which is without dotted circle in this case, Meera, Rachana and some other fonts has a trick. These fonts contains a special glyph ഞ. This glyph is mapped to an akhand rule, see Figure 8.12.

With this akhand rule, we will get expected rendering for "nth" patterns as shown in Figure 8.13.

You might have a question why dot is appearing for anusvara since that also normally applied to a consonant cluster. But there are cases anusvara applied to vowel signs as shown in figure 8.14.

$$o_o = \circ + \circ o + \circ o$$

Figure 8.12: Akhand rule for oo. Dotted circle is included in the rule.

16-oo

Figure 8.13: 16th with oo rendered correctly.

നാലുoo, അടീoo, ധൂoo, ധിoo,  
റാoo, സ്റ്റൂoo, സ്റ്റൂoo

Figure 8.14: Anusvara oo applied to a vowel sign.

Extra elongation for vowel signs

Repeated vowel signs are used to denote elongation of a vowel pronunciation. But rendering engines used to fail to render them properly because if vowel signs are repeating, from second vowel sign onwards it is not attached to a consonant cluster and it causes dotted circle. Now a days, Harfbuzz rendering engine does not put dotted circle in between them. See Figures 8.15 and 8.16.

#### 8.4.4 Samvruthokaram

Samvruthokaram is a mid vowel sound representation for vowel sign u. Now a days, it is denoted just using virama like അത് ഉണ്ട് , also known as Pseudo Samvruthokaram <sup>1</sup> But using vowel sign of u and virama-് to represent this is not rare.

As you see, the vowel sign u attach to the consonant and virama appears on top. With modern orthography the rendering is different, with different u and virama signs explicitly shown as shown in Figure 8.18. Samvruthokaram rendering in traditional orthography is shown in Meera font in Figure 8.17 for a comparison.

<sup>1</sup>Chandrakkala. Samvruthokaram. Chillaksharam. From the perspective of Malayalam Collation R. Chitrajakumar and N. Gangadharan Rachana Akshara Vedi <http://www.unicode.org/L2/L2005/05210-malayalam.pdf>

ടാടാടാ

Figure 8.15: Repeated vowel signs.

ടാടാടാഃടാടാടാ

Figure 8.16: Repeated vowel signs, dotted circle start appearing after 4th vowel sign.

അത് ഉണ്ടു്

Figure 8.17: Rendering of Samvruthokaram with Meera font.

അതു് ഉണ്ടു്

Figure 8.18: Rendering of Samvruthokaram with Raghu Malayalam font.

It will look a bit odd to see these two signs in rendering in modern orthography. In modern orthography samvruthokaram is very rare. Because of this there was an attempt to add a glyph like virama to denote Samvruthokaram in fonts like Raghu Malayalam and Lohit. But reverted recently(2013) since it was quite an experiment. <sup>2</sup>

### History of Samvruthokaram rendering

Samvruthokaram was one of the difficult rendering to get working in the initial days of Malayalam computing. Rendering engines had the general concept that the u vowel sign will not join with a virama. Of course Samvruthokaram was an exception to that rule. Till 2009, this problem continued. Rendering engines displayed samvruthokaram with dotted circled around Virama. Volunteer developers from Swathanthra Malayalam Computing project worked closely with Pango<sup>3</sup>, Qt<sup>4</sup>, ICU<sup>5</sup> rendering engines to get the bug fixed.

<sup>2</sup>Red Hat bugzilla:Samvruthokaram ligature is wrong [https://bugzilla.redhat.com/show\\_bug.cgi?id=1013183](https://bugzilla.redhat.com/show_bug.cgi?id=1013183)

<sup>3</sup><http://www.pango.org/>

<sup>4</sup><http://qt-project.org>

<sup>5</sup><http://userguide.icu-project.org/layoutengine>

Bug reports in respective bug tracking systems gives a detailed history of this effort.

1. Red Hat bug: [https://bugzilla.redhat.com/show\\_bug.cgi?id=242016](https://bugzilla.redhat.com/show_bug.cgi?id=242016)
2. Pango Bug [https://bugzilla.gnome.org/show\\_bug.cgi?id=504810](https://bugzilla.gnome.org/show_bug.cgi?id=504810)
3. ICU Bug <http://bugs.icu-project.org/trac/ticket/6108>

### 8.4.5 Conjunct Signs for ഐ, റ, ല

Three consonants ഐ, റ, ല has vowel like symbols in Malayalam. These signs attach to the previous consonant cluster just like vowel signs.

ക് + ഐ = ക്കൈ

ക് + റ = ക്ര

ക് + ല = ക്കല്ല

Figure 8.19: Rendering of consonant signs with Meera font.

ക് + ഐ = ക്കൈ

ക് + റ = ക്ര

ക് + ല = ക്കല്ല

Figure 8.20: Rendering of consonant signs with Raghu Malayalam font.

### 8.4.6 Reph

### 8.4.7 Dot Reph

Dot-Reph is used to represent Malayalam letter റ in its dead form (or without vowel form), as a dot over the consonant following it. It is denoted by a dot on top of the letter. Dot reph is also known as Gopi Reph. Usage of Dot Reph is not popular nowadays compared to the texts in past decades.



Figure 8.21: Dot Reph

The glyph can be circle shaped or in the form of a water drop upside down (also known as a Gopi sign).

കർണ്ണൻ - കണ്ണൻ

Figure 8.22: Rendering of word കർണ്ണൻ without and with dotreph

In the above example, the word meaning remains same in both forms. The Dot Reph was not encoded till Unicode version 5.1.

One of the reasons for not having an atomic code point in initial versions is that it was considered as a typographer's choice to use it as a glyph variant for Chillu R.

In Unicode 5.1 this was encoded atomically with a code point 0D4E. As per the proposal<sup>6</sup>, the reason for encodings were given as

1. റ + റ് + ZWJ is the natural choice for Dot Reph, but already used for MALAYALAM CHILLU RR before Unicode 5.1. So if somebody want to use both Chillu form and Dot Reph, it is impossible.
2. If റ + റ് is used for Dot Reph, it will conflict with the usual usage of that sequence for words like ഭാര്യ സൂര്യൻ, കാര്യം etc will break.
3. If റ + റ് is used for Dot Reph, In the words like ഭാര്യ സൂര്യൻ, കാര്യം etc, it is that forms the conjuncts and not റ.

<sup>6</sup> Dot Reph encoding proposal <http://std.dkuug.dk/jtc1/sc2/wg2/docs/n3676.pdf>

Swathanthra Malayalam computing had opposed this proposal in the document they submitted to Unicode <sup>7</sup> with alternate way of representing Dot Reph without introducing a code point difference to words with and without Dot Reph, as in the കർണ്ണൻ example. Their proposal was to use Dot Reph if a font maps റ + ് to a Dot Reph glyph or fallback to റ if the font does not have the glyph. But this was not accepted and in Unicode 5.1, Dot Reph was encoded.

The Dot Reph appears as a dot above the subsequent Malayalam consonant or cluster.



Figure 8.23: Rendering of റവ using Dot Reph

More examples of words with Dot Reph is given below.

ഭായു സൂര്യൻ കായും കർത്താവ് ആക് വത്തളം സുഹാർത്തോ

Figure 8.24: More Dot Reph examples

Rendering complexities of Dot Reph, with examples such as ആക് വത്തളം സുഹാർത്തോ.

Dot Reph is best implemented in font using Opentype Glyph Class 'Mark'. Base glyphs and conjuncts are not marked as 'Mark' class, but an Anchor point to those glyphs could be added indicating the position where Dot Reph would be placed.

Harfbuzz and Uniscribe reorders Dot Reph after the base glyph/conjunct. For instance hb-shape utility (from Harfbuzz source tree) shows the output of `തു <dotreph, th1, xx(virama), th1, u1(u-sign)>` as `[th1th1u1=0+2461|dotreph=0-575,-5+0]`. Thus, it is only necessary to add an Anchor point to glyph `th1th1u1` at the desired coordinate to render `dotreph`.

An example of how to do this using Fontforge:

### 8.4.8 Chillus

<sup>7</sup>Atomic chillu's are Unacceptable L2/08-038 [http://wiki.smc.org.in/images/2/23/SMC\\_Unicode\\_5.1.pdf](http://wiki.smc.org.in/images/2/23/SMC_Unicode_5.1.pdf)



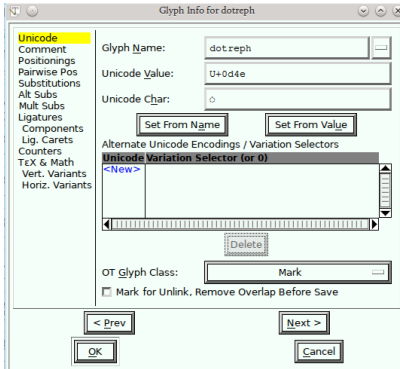


Figure 8.25: Set OT Glyph Class of dot:reph as 'Mark'

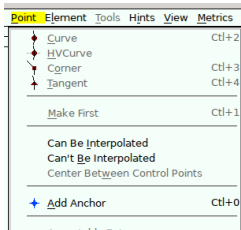


Figure 8.26: Add Anchor point menu

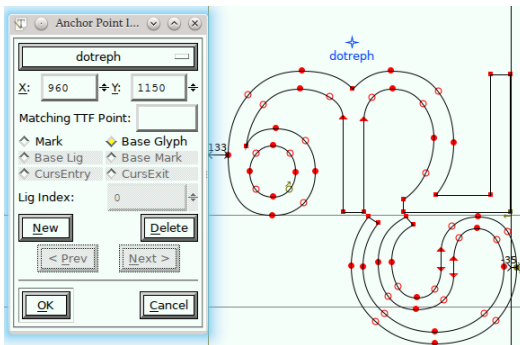


Figure 8.27: Position the dot:reph Anchor point on the glyph

## 8.4.9 Stacking

- 8.4.10 Font metrics
- 8.4.11 Positioning rules
- 8.4.12 ZWNJ and ZWJ Signs
- 8.4.13 Prebase substitutions
- 8.4.14 Akhand forms
- 8.4.15 Below base forms
- 8.4.16 Below base substitutions
- 8.4.17 Half forms
- 8.4.18 Postbase forms
- 8.4.19 Latin glyphs and punctuations
- 8.4.20 Kerning
- 8.4.21 Shape references
- 8.4.22 Left and right bearings
- 8.4.23 Italic variant
- 8.4.24 Bold variant

## 8.5 Design

### 8.5.1 Number of glyphs

The main difference between traditional and new orthography is the number of glyphs to be designed and drawn for a font. Traditional orthography fonts require more than 1000 glyphs while new orthography fonts need less than 400 glyphs.

Meera font has around 1100 glyphs while Raghu Malayalam font has around 350 glyphs.

## 8.5.2 Guidelines

There is no rule about whether a particular glyph can be present only in traditional font or new orthography font. It is up to the designer. But it is possible to list certain guidelines about this.

1. Include basic punctuations. Punctuations can come inside Malayalam text. To make the rendering consistent, punctuation marks that match the font style should be present in the font.
2. Include Arabic numbers
3. All Malayalam characters encoded by the current Unicode version
4. Zero width joiner Zero width non-joiner place holder glyphs - TODO explain in detail
5. A fall back glyph mapped to .notdef, usually a box
6. Dotted circle - To denote invalid combining marks, ie usually invalid vowel signs at wrong positions, opentype specification recommends to add a glyph of Unicode character U+25CC.<sup>8</sup> The glyph should be a dotted circle. A few examples where they will be used is given below.



Figure 8.28: Dotted circle example

Rendering engines can decide whether a vowel sign is at valid position or not and implementation may vary. One should not expect linguistic correctness on this. For example, a vowel sign after a vowel is invalid for Malayalam, but following rendering does not show dotted circle.

7. Malayalam glyphs with opentype feature tables

---

<sup>8</sup>Invalid combining marks. <http://www.microsoft.com/typography/OpenTypeDev/malayalam/intro.htm>

# ആം ഉയ്യൂ

Figure 8.29: Rendering not showing dotted circle even though incorrect in linguistic sense.

8. Rendering consistency - It will be easy to explain this with an example. Consider the case of ്. The ka is stacked under ya. If a font has this kind of stacking, the same font must have a number of additional glyphs to make rendering consistent.



Figure 8.30: Consistent stacking of letters

If the designer miss to add ് ് ്, the consistency will be broken and it will render as follows:



Figure 8.31: Inconsistent stacking of letters

It will also get broken if the font has ് but missed to add glyphs for ് ്.

If the font designer decides not to have glyph for ് at all, none of the above issues arise, but ് is a common ligature in Malayalam.

In the last example, rendering is not broken if the font is a new orthography font. But if it is a traditional orthography font, designer should make sure all stacked glyphs are present in the font to provide consistent rendering. This is one of the reason for lot of glyphs appearing in a traditional orthography font.

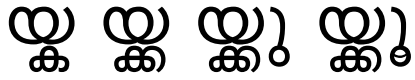


Figure 8.32: Inconsistent stacking of letters

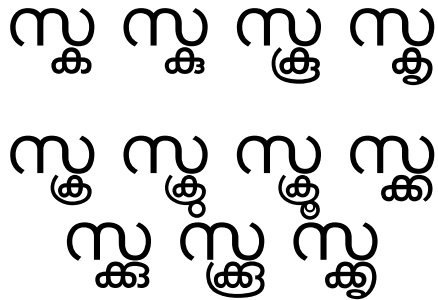


Figure 8.33: All glyph combinations of sa and ka stacked - from Meera font



9

---

Odiya

---





10

---

Panjabi

---



11

---

Tamil

---



## Telugu

---

### 12.1 Introduction

Telugu script, an abugida from the Brahmic family of scripts, is used to write the Telugu language, a language found in the South Indian state of Andhra Pradesh as well as several other neighboring states. It gained prominence during Vengi Chalukyan era.



13

---

## Glossary

---





---

## List of Figures

---

2.1	Glyphs inside Roboto font . . . . .	6
2.2	Glyphs inside Meera font . . . . .	7
2.3	Glyphs inside Lohit Bengali font . . . . .	8
2.4	The Devanagari ddhrya-ligature (द् + ध् + र् + य = द्धय)	8
2.5	The Malayalam kthu-ligature (ക + ൾ + ത + ൹ ) . . . . .	9
2.6	The Gujarati 's' formed from SA and ANUSVARA . . . . .	9
8.1	Text rendering using traditional orthography. Font used is Meera . . . . .	22
8.2	Text rendering using modern orthography. Font used is Lohit Malayalam . . . . .	23
8.3	Rendering difference of u and uu vowel signs . . . . .	24
8.4	Rendering difference of REPH sign . . . . .	25
8.5	Malayalam vowel signs. . . . .	25
8.6	Malayalam vowel ീ getting applied. . . . .	26
8.7	Malayalam vowel െ getting applied after reordering. . . . .	26
8.8	Malayalam vowel െ getting spit and applied at left and right of ഴ . . . . .	26
8.9	Split matra glyphs inside Meera font for െ, െ and െ	27
8.10	Wrong rendering of Samvruthokaram. . . . .	27
8.11	Correct rendering of Samvruthokaram. . . . .	27
8.12	Akhand rule for െ. Dotted circle is included in the rule.	28
8.13	16th with െ rendered correctly. . . . .	28
8.14	Anusvara െ applied to a vowel sign. . . . .	28
8.15	Repeated vowel signs. . . . .	29
8.16	Repeated vowel signs, dotted circle start appearing af- ter 4th vowel sign. . . . .	29
8.17	Rendering of Samvruthokaram with Meera font. . . . .	29

8.18	Rendering of Samvruthokaram with Raghu Malayalam font. . . . .	29
8.19	Rendering of consonant signs with Meera font. . . . .	30
8.20	Rendering of consonant signs with Raghu Malayalam font. . . . .	30
8.21	Dot Reph . . . . .	31
8.22	Rendering of word without and with dotreph	31
8.23	Rendering of ൠ using Dot Reph . . . . .	32
8.24	More Dot Reph examples . . . . .	32
8.25	Set OT Glyph Class of dotreph as 'Mark' . . . . .	33
8.26	Add Anchor point menu . . . . .	33
8.27	Position the dotreph Anchor point on the glyph . . . . .	33
8.28	Dotted circle example . . . . .	35
8.29	Rendering not showing dotted circle even though incorrect in linguistic sense. . . . .	36
8.30	Consistent stacking of letters . . . . .	36
8.31	Inconsistent stacking of letters . . . . .	36
8.32	Inconsistent stacking of letters . . . . .	37
8.33	All glyph combinations of sa and ka stacked - from Meera font . . . . .	37

---

## References

---

Converting a legacy ASCII based Indic font to an Unicode compliant Open Type Font: [http://sayamindu.randomink.org/conv\\_guide/](http://sayamindu.randomink.org/conv_guide/)

Pothana Paper: <http://upload.wikimedia.org/wikipedia/te/c/c2/Pothanapaper.PDF>

State of Text Rendering By Behdad Esfahbod <behdad behdad org>:  
<http://behdad.org/text/>



15

---

## Appendices

---

[Editor notes]