

# Unicode IPA Release Notes

*Martin Hosken,  
SIL Non-Roman Script Initiative (NRSI)*

## Introduction

With the advent of Unicode, the opportunities for those using the IPA are immense. Many of the previous technological difficulties of being limited to 8-bit data seem to have a solution in Unicode. Now we can have one font containing all of IPA, standard ASCII and perhaps anything else we want.

This font kit contains a font and supporting files covering some 390 codepoints with 530 glyphs.

This document describes each of the components of the kit in more detail.

## Contents

For a font like this, just a font on its own is of limited value. Thus the font kit contains a number of supporting files.

## Font

The font is based on the SIL Doulos regular face and covers all the characters in these character sets:

- The older SILIPA93 font kit,
- The Western European Codepage 1252 (Latin-1, including ASCII)
- The Western European Mac Codepage.

The font is designed to work with three different rendering technologies:

- |                 |                                                                                                                                                                                                                                                                                                                                                                   |
|-----------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>TrueType</b> | Simple TrueType rendering, using one codepoint per glyph. The resulting rendering is far from satisfactory since the font is designed to map directly to Unicode, resulting in only one codepoint for each diacritic regardless of required position. No presentation forms have been included to account for the positional variants from the SILIPA93 encoding. |
| <b>OpenType</b> | The font supports smart rendering using OpenType™ technology. This allows an OpenType aware application to use features within the font to display IPA correctly. This includes: diacritic positioning, tone letter ligating and dotless i resolution.                                                                                                            |
| <b>Graphite</b> | Graphite is a smart rendering technology from SIL <sup>1</sup> which allows Graphite aware applications to render IPA correctly with correctly positioned diacritics, tone letter ligation, dotless i resolution and correct positioning of double diacritics (over-arch).                                                                                        |

---

<sup>1</sup> See <http://www.sil.org/computing/graphite> for more details

For more details about the capabilities of the font regarding which characters are supported and how the various technologies are supported by the font, see the Technical Details section later in this document.

Users may find using the font as a far from satisfying experience, at least initially. This is because without the newer rendering technologies, there is no way for an application to position diacritics correctly directly from Unicode encoded data. An alternative would be to add positional variants into the Private Use Area, but this would result in non-standard encoded data which would cause even greater problems in the long run. In short, the font is a modern smart font.

## Keyman

Keyman<sup>2</sup> is a keyboard definition utility that allows for complex keyboard definitions. Two keyboard definitions are included with this font kit. `ipa93a_5.kmx` is a keyboard that generates only ANSI codes according to the SILIPA93 encoding, including correctly positioning diacritics (within the constraints of the font) and dotless *i* resolution.

`ipaUni10.kmx` is a keyboard definition which enters Unicode values for IPA using the same keyboard layout as for `ipa93a_5.kmx`. Details of the keyboard layout may be found in the section on Keyboard Layout.

## Conversion

The conversion description file consists of an XML description based on the Unicode Technical Report UTR22. The language has been extended to support the contextual nature of the conversion, particular from Unicode to bytes. Also enclosed is a compiled form of this conversion table for the SILtec engine.

## Keyboard Layout

The keyboard definitions for Unicode and ANSI output are not identical, but they are very similar. Due to the larger character set supported by the Unicode keyboard, the Unicode keyboard has the following differences:

- Upper case keys produce upper case letters rather than as speed keys to IPA characters
- Numbers produce numbers, unless typing a sequence of tone letters.
- Numbers and hyphen followed by a circumflex (^) key result in a superscript number or hyphen.

The keyboard layout is described in terms of an IPA chart rather than a keyboard. This is because many base characters are typed as a sequence of a letter followed by one of <, > or = which are characters used to modify a base character to another base character. Diacritics are typed as sequences of an appropriate key. It is recognized that this is far from satisfactory, particularly in a Unicode context and suggestions for improvement are welcomed.

---

<sup>2</sup> See <http://www.tavultesoft.com> for more details of this product.

	Bilabial	Labio-dental	Dental	Alveolar	Post-alveolar	Retro-flex	Palatal	Velar	Uvular	Pharyngeal	Glottal
Plosive	p p b b		t t{ d d{	t t d d		ʈ t< ɖ d<	c c ɟ j=	k k g g	q q ɢ G=		ʔ ʔ=
Nasal	m m	ɱ m>	ɳ n{	n n		ɳ n<	ɲ n=	ŋ n>	ɴ N=		
Trill	ʙ B=			r r					ʀ R=		
Flap				ɾ r>		ɽ r<					
Fricative	ɸ f= β b=	f f v v	θ t= ð d=	s s z z	ʃ s= ʒ z=	ʂ s< ʐ z<	ç c= j j<	x x ɣ g=	χ x= ʁ R>	ħ h> ʕ ʔ<	h h ɦ h<
Lateral Fricative				ɬ l= ɮ l>							
Approx.		ʋ v=		ɹ r=		ɻ R<	j j	ɰ w>			
Lateral Approx.			ɭ l{	ɭ l		ɮ l<	ʎ L<	ɮ L=			
Implosive	ɓ p> ɗ b>			ɮ t> ɠ d>			ɕ c> ɟ j>	ɟ k> ɣ g>	ɢ q> ɢ G>		

ʍ	w=	VL Labial-Velar Approximant
w	w	Vd. Labial-Velar Approximant
ɥ	h=	VL Labial-Palatal Approximant
ç	c<	VL alveolopalatal fricative
ʐ	z>	Vd. alveolopalatal fricative
ɧ	H>	Simultaneous S and x
ɺ	L>	Vd. alveolar lateral flap

⦿	p=	Bilabial click
ǀ	!<	Dental click
ǃ	!	(Post-)alveolar click
ǁ	!=	Palatoalveolar click
ǁ	!>	Alveolar lateral click
ǀ	Q=	Vd. epiglottal Plosive
ʜ	H=	VL epiglottal fricative
ʕ	Q<	Vd. epiglottal fricative

Vowels	Front		Central		Back	
Close	ɪ i	y y	ɨ I=	ʉ U=	ɯ u=	u u
Near-close	ɪ i=	ʏ y=				ʊ u<
Close-mid	e e	ø o>			ɤ O>	o o
Mid			ə e= ɜ e>	ɵ O=		
Open-mid	ɛ e<	œ E<			ʌ u>	ɔ o<
Near-open	æ a<		ɐ a>			
Open	a a	ɶ E>			ɑ a=	ɒ o=



<b>U+00A0 – U+00FF:</b>	All characters
<b>U+0100 – U+017F:</b>	The following characters: U+0127, U+0131, U+014B, U+0152, U+0153, U+0161, U+0161, U+0178, U+017D, U+017E.
<b>U+0180 – U+024F:</b>	The following characters: U+0192, U+01C0 – U+01C3.
<b>U+0250 – U+02AF:</b>	All characters except: U+025A, U+025D, U+0269, U+0277, U+027C, U+027F, U+0285 – U+0287, U+0293, U+0296, U+0297, U+029A, U+02A0, U+02A3 – U+02AF.
<b>U+02B0 – U+02FF:</b>	The following characters: U+02B0, U+02B2, U+02B7, U+02BC, U+02C6 – U+02C8, U+02CC, U+02D0, U+02D1, U+02D6 – U+02DE, U+02E0, U+02E1, U+02E3 – U+02E9.
<b>U+0300 – U+036F:</b>	The following characters: U+0300 – U+0304, U+0306, U+0308, U+030A – U+030C, U+030F, U+0316 – U+031A, U+031C – U+0320, U+0324, U+0325, U+0329, U+032A, U+032C, U+032F, U+0330, U+0334, U+0339 – U+033D, U+0361.
<b>U+0370 – U+03FF:</b>	The following characters: U+03A9, U+03B2, U+03B8, U+03C0, U+03C7
<b>U+2000 – U+206F:</b>	U+2013 – U+2044 excepting: U+2015, U+2017, U+201B, U+201F, U+2023 – U+2025, U+2027 – U+202F, U+2031 – U+2038, U+203B – U+203E, U+2040 – U+2043.
<b>U+2070 – U+209F:</b>	The following characters: U+2070 – U+2079, U+207B, U+207F.
<b>U+20A0 – U+20CF:</b>	The following character: U+20AC
<b>U+2100 – U+214F:</b>	The following character: U+2122
<b>U+2190 – U+21FF:</b>	The following characters: U+2191, U+2193, U+2197, U+2198
<b>U+2200 – U+22FF:</b>	The following characters: U+2202, U+2206, U+220F, U+2211, U+221A, U+221E, U+222B, U+2248, U+2260, U+2264, U+2265.
<b>U+25A0 – U+25FF:</b>	The following character: U+25CA.
<b>U+FB00 – U+FB4F:</b>	The following characters: U+FB01, U+FB02.

## Private Use Area

To provide backward compatibility with the SILIPA93 encoding, the following characters are included as Private Use characters:

U+F8FA	Superscript eng (U+014B)
U+F8FB	Superscript nya (U+0272)
U+F8FC	Superscript m (U+006D)

## Graphite Support

The Graphite tables in the font provide support for the following behaviours:

- Dotted i, j, barred i all lose their dot in the presence of an upper diacritic
- Diacritics are correctly positioned according to the width of the base character, by character rather than by width class (i-width, o-width, etc.)
- Diacritics can stack to any depth although stacking behaviour is designed assuming Unicode canonical ordering (i.e. lower diacritics occur before upper diacritics) with a maximum of 4 lower diacritics. Non-canonical orderings are supported, but are not guaranteed.

- Tone letters (as opposed to diacritics) ligate up to a sequence of 3 different tones (then a new letter starts). Since tone letters give no indication of timing, sequences of the same pitch are concatenated into a sequence of length one. Also glides between pitches in a straight line are concatenated into a single glide between the two outer pitches.
- Double acting diacritics (U+0361) automatically position above the highest diacritic on either side.
- fi and fl are converted to ligatures when there are no diacritics associated with either of the component characters.

In addition the font has a feature called: Pitches as Numbers with an id of 23. If this feature is set to a value of 1 then a sequence of tone letters are rendered as a sequence of superscript numbers separated by superscript hyphens.

## Diacritics

Here we list which characters are which types of diacritic

<b>Upper diacritic</b>	U+0300 – U+0304, U+0306, U+0308, U+030A – U+030C, U+030F, U+033D, U+0361
<b>Below diacritic</b>	U+0316 – U+0319, U+031C – U+0320, U+0324, U+0325, U+0329, U+032A, U+032C, U+032F, U+0330, U+0339 – U+033C
<b>Centre diacritic</b>	U+0334
<b>Rhotic hook</b>	U+02DE

Notice that although the rhotic hook is considered a spacing modifier, it is rendered as though it were combining. Likewise U+031A is a combining character which is rendered as a spacing modifier letter.

Here are the lists of which character may be combined with a particular type of diacritic

<b>Upper diacritic</b>	U+0041 – U+005A, U+0061 – U+006A, U+00C0 – U+00D6, U+00D8 – U+00F6, U+00F8 – U+02A2, U+03B2, U+03B8, U+03C7, and all upper diacritics
<b>Below diacritic</b>	U+0041 – U+005A, U+0061 – U+006A, U+00C0 – U+00D6, U+00D8 – U+00F6, U+00F8 – U+02A2, U+03B2, U+03B8, U+03C7, and all below diacritics
<b>Centre diacritic</b>	U+0064, U+006C, U+0072 – U+0074, U+007A, U+00F0, U+0161, U+017E, U+027E, U+0283, U+0292
<b>Rhotic hook</b>	U+0061, U+0065, U+0069, U+006F, U+0075, U+0079, U+00E0 – U+00E6, U+00E8 – U+00EF, U+00F2 – U+00F6, U+00F8 – U+00FD, U+00FF, U+0131, U+0153, U+0250 – U+0252, U+0254, U+0258 – U+025E, U+0264, U+0268, U+026A, U+0275, U+0276, U+0289, U+028A, U+028C

## OpenType

The OpenType support is similar to the Graphite support. The behaviours supported are:

- Dotted i, j and barred i are converted to dotless form preceding an upper diacritic
- Diacritics are positioned correctly and stack according to a similar behaviour in Graphite. The same tables apply to the OpenType behaviour.
- Tone letters ligate up to a strict maximum of 3 characters regardless of the content of those characters. Thus a sequence of identical codes will not ligate.
- Double acting diacritics are positioned with default positioning only.

The behaviour is enabled for the ‘latn’ script and the ‘dflt’ and ‘IPA ’ languages. Within these languages, the following features are supported:

GSUB:	ccmp	Contextual replacement of i, j and barred i by dotless forms
	liga	True letter ligation. This means that the ligation can be turned off
GPOS:	mark	Attaching diacritics to base characters
	mkmk	Diacritic stacking for upper and below diacritics

## Conclusion

The SILDoulosUnicode IPA font and supporting files provides a Unicode font covering the basic IPA requirements for the IPA93 standard. It is an early example of a new family of fonts based around smart rendering technologies. Future work aims to extend this font into other faces and also to provide other fonts with extended character sets to provide for the needs of linguistic analysis.

If you have any comments regarding this font kit, please contact me at [mailto:martin\\_hosken@sil.org](mailto:martin_hosken@sil.org).

Martin Hosken

July 2001