UTC Document Register **L2/14-065**

A bag of suggested improvements to Unicode's provisional Indic properties

Roozbeh Pournader and Behdad Esfahbod, Google Inc.
February 5, 2014

Background

The Indic properties provided in the Unicode data files IndicMatraCategory.txt and IndicSyllabicCategory.txt have been extremely useful to the work on HarfBuzz, the open source text rendering engine used in Linux, Android, Chrome, and Firefox among several other systems. We have been able to leverage that data to support both major and minor scripts of South and Southeast Asia, sometimes without much knowledge of the scripts themselves. As such, we strongly support the promotion of that data to informative properties, especially if the data is extended and completed to be more useful for detecting and matching parts of Indic syllabic structures.

Still, we believe that improvements could be made to both the properties and their values, and we would like to ask the authors to consider such changes. The present document is a mixed bag of suggestions and wishes about that data by the authors, mostly based on Behdad Esfahbod's early notes in developing HarfBuzz's support for South and Southeast Asian scripts, drafted and compiled by Roozbeh Pournader. They should all be taken with a grain of salt, since, although with a couple of years of difference, the authors were still students of the intricacies of Indic writing systems when they wrote this.

The authors urge the members of the Unicode Technical and Editorial committees to consider these in further improvements to the Indic properties. These would help in the further development of Unicode-conformant rendering engines, fonts, OCR software, and other similar applications.

Suggestions regarding the Indic Syllable and Matra categories

1. Similar to bidi and Arabic joining properties, where special formatting characters have their own property values, it is extremely useful to add special property values for ZWJ and ZWNJ in Indic_Syllabic_Category. Such one-member property values would make it possible for algorithm using the property to not look at codepoints directly.

2. It is very useful to add characters that otherwise participate in Indic joining to the existing set defined to have the property value Consonant_Placeholder. Candidates for such additions include the Kannada placeholders at U+0CF1 and U+0CF2 (and perhaps their Vedic counterparts), the Myanmar symbol for "aforementioned" at U+104E, and various digits and

punctuation, which are used in Malayalam and very possible other Indic scripts. For example, according to TUS Core Specification 6.2, section 9.9, page 319:

> "More generally, rendering engines should be prepared to handle Malayalam letters (including vowel letters), digits (both European and Malayalam), dashes, U+00A0 no-break space and U+25CC dotted circle as base characters for the Malayalam vowel signs, U+0D4D malayalam sign virama, U+0D02 malayalam sign anusvara, and U+0D03 malayalam sign visarga."

3. The Invisible class of characters defined in Indic Matra Category appears to be really a subdivision of viramas and as such, may better be moved to Indic Syllabic Category.

4. Further subdivide the Consonant category to have a special class for Ra in scripts where it has special visual behavior, in particular, scripts where the sequence <Ra, Virama> or <Ra, Virama, ZWJ> makes a reph. This is important for understanding the syllabic structure of Indic text, since in such scripts sequences such as <Ra, Virama, Dotted Circle/NBSP> are valid syllables, where other consonants wouldn't be. With such a subdivision, we can detect Ra/Reph's without needing to hard-code per-script codepoints in rendering engines or OCR software.

   In a way, this is really a Reph category for a sequence of characters instead of one character, but as we can't do that with the existing structure, we can put these Ra's in a subcategory that would point to the Reph behavior.

   It would be even more useful if we could separate the scripts where the Reph is formed by <Ra, Virama, ZWJ> (as in Sinhala and Telugu) as opposed to <Ra, Virama> (most other scripts).

5. Divide the Consonant_Repha category into the cases where the Repha is used in logical order (such as Malayalam), as opposed to visual order (Khmer, Javanese, etc). Currently, the general category of the character can be used as a hint for making a distinction, but we believe the distinction is large enough and the hint cannot be assumed to work for future characters.

6. Assign syllabic and matra categories to the Khmer characters at U+17CB, U+17CD .. U+17D0, and U+17D3. Visually, they act like top matras.

7. Consider subdividing the virama category for characters that are linguistically "killer"s, but visually act like matras, or moving such characters to a matra set. An example would be the Khmer Viriam at U+17D1, that could be treated like a normal matra for Indic categories, like the Javanese killer U+A9C0.

8. Add Indic properties for the Gurmukhi Addak at U+0A71 that currently doesn't have any Indic properties. This is what we spotted as missing, but as there may be more characters missing, it would be a good idea if the maintainers of the properties could check for more potentially missing characters from the properties.

9. Make information about non-matra characters that are reordrant available as part of the Indic properties. Some of these, listed in Table 4-4 of the Core Specification, are the Tai Tham U+1A55, the Lepcha U+1C34 and U+1C35, and the Cham U+AA34.

10. Provide information about the visual components of matras in split categories that don't have canonical decompositions to their pieces through some additional property, potentially also including characters that ***do*** have canonical decompositions. These would be very useful in both rendering text and OCR applications. Here is a first list, created by Jonathan Kew and Roozbeh Pournader:

    0AC9 => 0AC5 0ABE (Gujarati Candra O)
    0F77 => 0FB2 0F81 (Tibetan Vocalic RR)
    0F79 => 0FB3 0F81 (Tibetan Vocalic LL)
    17BE => 17C1 17B8 (Khmer OE)
    17C4 => 17C1 17B6 (Khmer OO)
    1925 => 1920 1923 (Limbu OO)
    1926 => 1920 1924 (Limbu AU)

    Note that the list is not complete, as not all individual pieces are not encoded, as in the following characters:

    17BF => 17C1 ???? (Khmer YA)
    17C0 => 17C1 ???? (Khmer IE)
    17C5 => 17C1 ???? (Khmer AU)
    1B3C => 1B42 ???? (Balinese La Lenga)

    Interestingly, Uniscribe (and compatible engines) implement all the ***five*** above Khmer characters (having or not having visual pieces encoded) by inserting an extra U+17C1 (their left piece) on their left side, and then expecting the character itself to just have the shape for the right part (or possibly be transformed to the shape of the right part, using the 'pstf' feature of GSUB tables, that some fonts do).

11. The Lepcha U+1C29 is listed as Top_And_Left in Indic Matra Category, but is listed together with Left matras in Table 4-4 of the Core Specification. As this is not a split character like the rest of the Top_And_Left set, a Matra Category of Left may be better for it for ***some*** rendering purposes. [See the Appendix at the end of the document for some comments from Peter Constable, Shriramana Sharma, and Suresh Kolichala.]
    This, and other similar non-split matras occupying more than one slot around a base letter, a further division or additional property or properties could be created. Such an additional property could point to the location of the primary position of the matra among the two positions listed. We believe the following could be a first such list:

    U+0B57: Top_And_Right, primarily Right (Oriya AU Length Mark)
    U+1C29: Top_And_Left, primarily Left (Lepcha OO)

U+A9C0: Bottom_And_Right, primarily Right (Javanese killer)
U+111BF: Top_And_Right, primarily Top (Sharada AU)

12. The Core Specification's Table 4-6 lists the Sinhala U+0DDD as "Left, Top, and Right", while its Indic Matra Category is Left_And_Right. We suspect the Core Specification is correct here, and the matra category should be changed to Top_And_Left_And_Right.

Fixes to the Core Specification

13. Table 4-4, on page 128 of the Core Specification, the Sinhala vowel sign at U+0DDA is mistakenly listed together with the characters of the left matra category. That is in contrast with its Indic Matra Category of Top_And_Left and its also being listed in Table 4-6 as "Left and top". It should be removed from Table 4-4.

14. Table 4-4 lists the Sharada U+11184 (independent AA) as reordrant. This is probably a typo for U+111B4 (dependent vowel I), as supported by U+111B4's matra category of Left. U+11184 should be corrected to U+111B4 in Table 4-4.

15. The Gujarati U+0AC9 is listed in the data file with the matra category Top_And_Right, but is missing from Table 4-6. It should be added there under "Top and right".

16. The Chakma U+1112E and U+1112F are listed in the data file with the matra category Top_And_Bottom, but are missing from Table 4-6. They should be added there under "Top and bottom".

17. Table 4-7 in the Core Specification is missing quite a few characters, all of which have combining class of zero and Indic Syllabic Category of Consonant_Subjoined:
    Tibetan: 0F8D, 0F8E, 0F8F
    Sundanese: 1BA2, 1BA3, 1BAC, 1BAD
    Lepcha: 1C24, 1C25
    Phags-pa: A867, A868, A871
    Javanese: A9BD

Bibliography

1. Behdad Esfahbod et al. 2014. *HarfBuzz, an OpenType text shaping engine*. http://www.harfbuzz.org/.
2. Google Inc. 2014. *The Noto open source fonts project*. https://code.google.com/p/noto/.
3. Srinidhi. 2013. "Representation of Jihvamuliya and Upadhmaniya in Kannada." UTC document register L2/13-242. http://www.unicode.org/L2/L2013/13242-kannada-rep.pdf.
4. The Unicode Consortium. 2013. "Unicode data file IndicMatraCategory.txt, version 6.3.0." http://www.unicode.org/Public/6.3.0/ucd/IndicMatraCategory.txt.
5. The Unicode Consortium. 2013. "Unicode data file IndicSyllabicCategory.txt, version 6.3.0." http://www.unicode.org/Public/6.3.0/ucd/IndicSyllabicCategory.txt.
6. The Unicode Consortium. 2013. *The Unicode Standard Version 6.2 – Core Specification*.

Appendix

Here is some quoted communication from the unicoRe mailing list, after an email by Behdad Esfahbod suggested a change to the matra category of the Lepcha U+1C29.

Behdad Esfahbod:

> "The standard lists U+1C29 as reordrant, but UCD marks it Top_And_Left. This is NOT a split glyph, so I think fixing UCD to mark it Left would be better."

Peter Constable, replying to Behdad Esfahbod:

> "I'm not sure I agree: you probably want classifications that tell you not only how characters need to reorder, but also classifications that indicate which structural matra positions relative to a base are occupied so as to recognize structurally-incoherent sequences in which two different characters compete for the same space.
>
> Keep in mind that the matras in these scripts do not behave like diacritics of European scripts, for which stacking behaviour can be expected. These marks are structurally like appendages to the base, and in several cases display as attached to the base; they do not stack. As a result, attempting to display two elements that go in the same location is not really viable; the only way to provide a legible result is to treat the sequence as exceptional, with the cluster breaking before the mark that resulted in the conflict."

Shriramana Sharma, replying to Peter Constable:

> "Hmm, I agree with this concept of marks competing for the same space.
>
> Just the day before yesterday I was wondering why in Kannada the VS for O ೊ is a combination of VS-E ೆ and VS-UU ೂ rather than the standard VS-E + VS-AA seen in all (most?) other Indic scripts. I then realized that it is probably actually the VS-AA ಾ which has been modified to be written from the bottom as ೂ to avoid spatial conflict with the VS-E ೆ which is also attached to the top right of the consonant... (Just my own ruminations, but it seems a valid conjecture. I'll later check it up if I can find any experts on this...)"

Suresh Kolichala, replying to Shriramana Sharma:

> "Your conjecture shows your Devanagari-centric thinking :-). Since the Dravidian languages makes a distinction between short and long vowels in /i/ and /o/ as well, Telugu-Kannada script had to devise two different combinations to represent them. The evidence from Telugu script is more explicit:
>
> mo (m + short-o) = ము = మ (me) + ు (shortu)
> మో (m + long-O) = మ (me) + ా (AA)
> yo (y + short-o) = యు = య (ye) + ు (short u)
> yO (y + long-O) = యో = య (ye) + ా (AA)

Kannada script diverged from this proto-form and used long-UU for short-o, while adding the standard length symbol for representing the long-O.

Why u? Remember that /u/, /o/ are phonologically related: both are rounded back vowels. Furthermore, there is an alternation between /u/ and /o/ in the south Dravidian languages(+Telugu). Eg: puga/poga, durai/dora, puDi/poDi etc.”