

TO : UNICODE TECHNICAL COMMITTEE
 FROM : VAISHNAVI MURTHY KODIPADY YERKADITHAYA (vaishnavimurthy@gmail.com)
 VINODH RAJAN (vinodh@virtualvinodh.com)
 DATE : 14th July 2021

SUBJECT : **REPLIES TO —**
Recommendations to UTC #167 April 2021 on Script Proposals &
<http://www.unicode.org/L2/L2021/21073-script-adhoc-rept.pdf>
Norbert Lindenberg's email with comments 10 April 2021

1. *It was noted that a few technical issues still remain in the Tulu-Tigalari proposal (see 9b. below). Norbert Lindenberg suggested changing the name of TULU-TIGALARI JOINER to TULU-TIGALARI CONJOINER. The Script Ad Hoc agreed. (The group also agreed the name CONJOINER would be a better name for Kawi, replacing KAWI SUBJOINER.)*

As suggested, Joiner is now re-named Conjoiner everywhere.

2. *This revised proposal notes recent updates to the proposal on page 2. The proposal states the repertoire supports use of the script for the Tulu, Kannada, and Sanskrit languages “for archival use,” as well as contemporary usage “without distorting the traditional orthography.” The proposal does not include some characters proposed by the Karnataka Tulu Sahitya Academy (such as the numbers and long and short /e/, /o/ and /ae/).*

The Tulu-Tigalari script proposal will cover only the traditional script's orthography in Kannada, Sanskrit and Tulu languages. This has been explicitly stated in the updated proposal as-well.

3. **REPHA**
In the proposal (page 34ff.), the ligated form of repha is invoked by repha with the conjoiner (proposed character name: TULU-TIGALARI JOINER), a novel approach that may be problematic. A conjoiner in Indic encoding is meant to connect two normal bases, while trying to conjoin this special base (repha) that essentially has a built-in conjoiner creates an irregularity in the encoding logic, and does not work with the Universal Shaping Engine (USE). If the approach described in the proposal is adopted, USE would need to be extended. ...

Since the use of the ligated repha is stylistic, does it need to be handled at the encoding level – could it be handled at the font-level? Or should ZWJ or ZWNJ be used in this case?

In our view, the authors should consider use of general joiners (ZWJ and ZWNJ) or a font-level approach for ligating repha. In order to make a decision, the authors may try comparing this ligation behavior with other unpredictable ligatures in the script and other scripts. The cost of

introducing a novel encoding logic is high, and a largely stylistic variation may not be worth it. Also they should consider that, in practice, fonts will only be able to provide a selective set of ligatures and will fall back to the non-ligated form. Such a fallback situation will lead to an unpredictable behavior of the conjoiner in a repha-ed akshara, similar to how virama's behavior is not fully predictable in the Devanagari encoding model.

Since the said sequence : Repha + Conjoiner + Consonant seems to break several rendering engines, the behaviour of the the conjoining Repha form has been re-looked-at. Ref. Section 5.5 c

4. LOOPED VIRAMA

The name (previously "ligating virama") appears to be acceptable.

The term "Ligating Virama" could be confused with a function. That is, a Virama that creates ligatures as pointed-out in one of the Ad-Hoc meetings. This Virama mark does not create ligatures, the "Conjoiner" forms ligatures according to our proposal. To avoid this possible confusion the names, "Looped Virama" was used instead.

5. CONJUNCTS

One difference between the "traditional" (manuscript) use of the script versus the modern "reformed" version is conjoining behavior, with conjuncts (known as "horizontal conjuncts" in the document) vs. subscripts (i.e., "vertical conjuncts") (Appendix 1, p. 49)...the use of the vertical versus the horizontal conjunct appears to be stylistic in the traditional orthography. Should they be handled at the encoding level? Note: Liang Hai also noted that the proposed specification of using ZWJ to "force" (horizontal) conjuncts also breaks the general assumption of Indic encoding models.

It is recommended to achieve this by applying features such as stylistic sets at the font level and not through control characters.

Tulu-Tigalari could handle subscript conjuncts like the encoded traditional Malayalam orthography, i.e., subscript is the default, and use font-level opt-in for horizontal conjuncts. However, this would make the orthography that only uses subscripts (vertical conjuncts) a secondary citizen. Would this be acceptable to modern users of the reformed orthography? This proposal has already been sent to the Tulu Sahitya Academy contacts. Those who wish to make specific comments on the proposal are encouraged to contact the proposal authors.

We already dealt with this in L2/21-092.

Norbert Lindenberg sent across an email containing a detailed list of comments to the authors on 10th April 2021. Below are replies to his comments.

6. *Page 2 and throughout: For Kawi, the Script Ad Hoc recommended to use “subjoiner” instead of “joiner” for compatibility with other scripts, despite Kawi having several post-base conjunct forms. I think in the meeting today we agreed that both Kawi and Tulu-Tigalari should switch to “conjoiner”.*

As suggested, Joiner is now re-named Conjoiner everywhere.

7. *Page 9: Figure 19 on page 65 seems to show alternate forms of vocalic rr and vocalic ll. Should these be discussed here?*

The Reph attaching with base vowels should not be restricted in this script. This is discussed under section 8.5 Repha.

8. *Page 9, 11, 12, 13: For Kawi, the SAH recommended to not encode characters atomically that can be represented equally well through character sequences. The reasoning was that, if two representations are possible, users may choose the wrong one, despite “do not use” lists in the proposal or the standard, and that it lets us avoid canonical decompositions. In your proposal this would clearly apply to letter ai, letter au, vowel sign ai, vowel sign oo, and vowel sign au. For letter ii and letter uu (alternate) it depends on whether the width of letter i and letter u usually includes the tail (in which case an added au length mark would end up noticeably far to the right) or whether it doesn't (in which case a combination with au length mark would look just like letter ii and letter uu (alternate)).*

We don't agree with this. Kawi users are familiar with the Javanese model. However, Indian users are used to have atomic code points for vowels. We are already deviating from the ISCII model with the use of conjoiner. Deviating too much is likely to confuse the users.

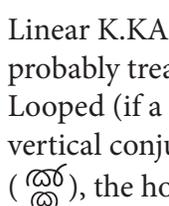
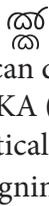
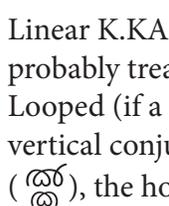
9. *Page 12: “Au length mark” seems to be used for more than just au, so maybe it should just be called “length mark”.*

It will be useful to call this AU length mark. The term "Length Mark" is commonly understood as the AA vowel mark. This "Au Length Mark" logic is carried forward from the Malayalam/Kannada orthography and it works well for those scripts.

10. Page 21, first example: *The virama after a post-base vowel is awkward to handle in the USE: While it treats pure killers as vowel signs, the USE expects above-base vowels before post-base vowels. This could be solved either by encoding the virama before the post-base vowel sign, or by overriding the syllabic category of the virama in the USE to a category that is expected later in the cluster, such as bindu. Either way, the proposal should discuss this issue and propose a solution. (I thought we had discussed this in an earlier SAH meeting, but don't see it in the draft minutes.)*

Thank you for pointing this out. Changing the input sequence is counter-intuitive. A USE override of the Virama's syllabic category to one that is expected later in the cluster will be required here.

- Page 22, last example: *In the USE, a joiner can't follow a pure killer. I don't know what to recommend here, seeing that the obvious alternative, [ka joiner ka looped_virama], is already taken. We should take this up in an SAH meeting with Andrew Glass.*

Linear K.KA + Looped Virama () and KA Looped + subjoined KA () should probably be treated as glyphic variants. i.e., KA + Conjoiner + KA + Virama can display KKA + Looped (if a font prefers horizontal conjuncts) or KA-looped + subjoined KA (if a font prefers vertical conjuncts). Since there is only one example found so far of the vertical representation (), the horizontal version () is recommended as a default for designing fonts.

$$\begin{array}{ccccccccc}
 \text{K.K} & / & \text{L.Vir} & = & \text{KA} & + & \text{Conjoiner} & + & \text{KA} & + & \text{L.Vir} \\
 \text{K.K} & & \text{L.Vir} & & \text{KA} & & \text{Conjoiner} & & \text{KA} & & \text{L.Vir}
 \end{array}$$

11. Page 23: *Is the dotted circle really necessary for unsupported ligatures? Wouldn't the lack of ligation speak for itself?*

It just makes it clear to the user that there is a mistake in the rendering. Its something that people are used-to seeing in the Indics. If the font designer so chooses, they can forgo the use of a dotted circle in their design.

12. Page 23: *Why does the treatment of J.JA (stylistic set) differ from that of K.KA (encoding with ZWJ) on page 21?*

This section has been revised to avoid this confusion.

13. Page 23: *No need to call out OpenType – Apple Advanced Typography also supports stylistic sets.*

Thank you for pointing this out. This correction has been made.

14. *Page 26, last example: shouldn't the character sequence include a virama?*

Thank you for pointing this out. This has been fixed.

15. *Page 34: a repha-joiner sequence isn't supported in the USE, as discussed in the SAH.*

This sequence is treated as a glyphic variant in the updated proposal that is to be handled at the font-level. This complication with the USE will therefore not arise.

16. *Page 35, second example: As for page 21 above.*

Same as reply to issue 10. above : Changing the input sequence is counter-intuitive. A USE override of the Virama's syllabic category to one that is expected later in the cluster will be required here.

17. *Pages 26, 35, 36: input sequence → character sequence. Keyboard input should be what users find natural, which might be repha after base.*

The input sequence is made to follow the Character Sequence everywhere in the updated proposal.

18. *Page 37: Is the visual placement of repha entirely up to the font designer?*

No. This has been corrected. The Repha is placed directly above a character/conjunct (centred). This placement is most commonly seen used in manuscripts and is the recommended form with all characters.

19. *Page 38: Is footnote 33 about keyboard input? "Writing or reading is subjective" seems wrong for the reading part – the vowel is always pronounced after the consonant. For writing, you should clarify whether you mean handwriting or keyboard input. For keyboard input, entering the vowel either before or after the consonants can be supported in modern software – it really depends on what feels more natural to users.*

The foot-note has been deleted as it doesn't add to the proposal as is creating unnecessary confusion. Thank you for pointing this out.

20. *Page 47: looped virama seems to be spacing, therefore Mn → Mc; NSM → L.*

Looped Virama is non-spacing. We'll retain the associated properties as such.

21. Page 47: The combining classes of Brahmic characters should generally be set to 0, except ccc=9 for virama-like characters and possible ccc=7 for nuktas. The order of characters within a Brahmic script cluster is generally based (loosely) on phonetic order, which conflicts with the visual logic of combining classes. (The encodings of Thai and Lao are separate messes.)

This has been fixed.

22. Page 47: The order of svarita and anudatta doesn't match the code chart.

The Svarita and Anudatta have character properties similar to their Devanagari/Vedic equivalents. The code chart has been changed to follow this order. Thank you for pointing this out.

23. Pages 47-48: Indic syllabic categories: 113BB..113C0 should be Vowel_Dependent # Mn. 113C2, 113C5 should be Vowel_Dependent # Mc. 113CE should be Pure_Killer # Mn. 113CF should be Pure_Killer # Mc. 113D3 should be Syllable_Modifier # Mn.

```
113B8..113C0 ; Vowel_Dependent # Mc [9] TULU-TIGALARI VOWEL SIGN AA..TIGALARI VOWEL SIGN VOCALIC LL
```

Is changed to :

```
113B8..113BA ; Vowel_Dependent # Mc [3] TULU-TIGALARI VOWEL SIGN AA..TIGALARI VOWEL SIGN VOCALIC II
113BB..113C0 ; Vowel_Dependent # Mn [6] TULU-TIGALARI VOWEL SIGN U..TIGALARI VOWEL SIGN VOCALIC LL
```

24. Page 48: Indic positional categories: 113B9..113BA should be Top_And_Right # Mc. 113D1 should be Top # Lo. 113D3 should be Bottom # Mn.

```
113B8..113BA ; Right # Mc [3] TULU-TIGALARI VOWEL SIGN AA..TULU-TIGALARI VOWEL SIGN II
```

Is changed to :

```
113B8 ; Right # Mc [3] TULU-TIGALARI VOWEL SIGN AA
```

Added :

```
# Indic_Positional_Category=Top_And_Right
113B9..113BA ; Top_And_Right # Mc [2] TULU-TIGALARI VOWEL SIGN I..TULU-TIGALARI VOWEL SIGN II
```

```
113D1 ; Top # Mn TULU-TIGALARI REPHA
```

Is changed to :

```
113D1 ; Top # Lo TULU-TIGALARI REPHA
```

113E1..113E2 ; Top # Mn [2] TULU-TIGALARI VEDIC TONE SVARITA.. TULU-TIGALARI VEDIC TONE ANUDATTA
Is changed to :

113E2 ; Top # Mn TULU-TIGALARI VEDIC TONE SVARITA

Add at the end of Indic_Positional_Category=Bottom:

113D3 ; Bottom # Mc TULU-TIGALARI SIGN TIDDU

113E1 ; Bottom # Mn TULU-TIGALARI VEDIC TONE ANUDATTA

25. *Editorial comments...*

All the editorial corrections suggested have been made. Thank you for sending them across.