Comments on Balinese Proposal, L2/05-008

Peter Constable, Microsoft
2005-2-1

The proposal to encode Balinese script is overall a well-prepared proposal that provides enough information to assess several important implementation issues. There is some information needed for implementations that was not included and which would have been useful, but does not have direct bearing on encoding decisions.[1] Of the various characters proposed, not all are individually identified in examples to demonstrate attested usage, and additional examples for certain cases would be welcomed as supplementary information.[2] Overall, though, there appears to no reason to question the claimed usage of the letterforms presented.

There are aspects of the proposal, however, that require special consideration. These have to do with decomposition mappings and canonical combining classes, two important aspects of normalization.

Also, the proposal discusses a particular spelling/sorting issue on page 3 and recommends that a certain character sequence, < RA, PEPET >, be displayed as though it were a different sequence, < RA REPA >. This will also be considered.

Multiple equivalent representations and decomposition mappings

The response provided in the proposal summary form to question 9 of section C, as to whether any of the proposed characters can be represented in terms of other proposed characters, requires further evaluation. As is common among Indic scripts, some vowel phonemes are written as multi-graphs involving two or more components that are also used individually. As a result, some of the proposed vowel signs have potential alternate representations in terms of character sequences. Specifically, several multi-graph vowels combine the letterform for /aa/ to the letterform for another vowel.

A summary of independent and dependent vowels is provided in the left columns of Table 1, with the use of /aa/ in multi-graph vowels is shown in red. If vowels are compared in terms of historic short/long pairs, it will be seen that many long forms are written like the corresponding short form with /aa/ added.

The right-hand columns of Table 1 show the minimal set of characters that would be required if no multi-graph forms are encoded as distinct characters. All of the multi-graph vowels shown in the left-hand columns of Table 1 can be displayed using sequences consisting only of characters in the right-hand columns.

---

[1] For instance, additional details regarding the rendering of various combinations of consonant clusters with vowel marks would be needed for font implementations.

[2] For instance, the caption to Figure 7 observes that the source in question does *not* make certain character distinctions. Examples from other sources demonstrating that these character distinctions are, nevertheless, attested would be helpful.

Vowel	Vowels proposed in L2/05-008 Independent	Dependent	Minimal set of required characters Independent	Dependent
a	ᬅ	--	ᬅ	--
aa	ᬆ	◌ᬵ	ᬆ	◌ᬵ
i	ᬇ	◌ᬶ	ᬇ	◌ᬶ
ii	ᬈ	◌ᬷ	--	◌ᬷ
u	ᬈ	◌ᬸ	ᬈ	◌ᬸ
uu	ᬈᬵ	◌ᬹ	--	◌ᬹ
voc. r	ᬉ	◌ᬺ	ᬉ	◌ᬺ
voc. rr	ᬊ	◌ᬻ	--	--
voc. l	ᬊ	◌ᬼ	ᬊ	--
voc. ll	ᬋ	◌ᬽ	ᬋ	--
e	ᬏ	◌ᬾ	ᬏ	◌ᬾ
ai	ᬐ	◌ᬿ	ᬐ	◌ᬿ
o	ᬑ	◌ᭀ	ᬑ	--
au	ᬒ	◌ᭁ	--	--
ae	--	◌ᭀ	--	◌ᭀ
oe	--	◌ᭀᬵ	--	--

Table 1. Proposed vowels, including multi-graph forms, and a minimal set of required characters

Therefore, it is clear that (the answer provided to question 9 of section C notwithstanding) some of the characters proposed *can* be represented using sequences of other proposed characters. While the UCS design principles indicate that, all other things being equal, characters that permit multiple equivalent representations are best avoided, there are certainly numerous precedents for encoding multi-graph vowels of Brahni-derived scripts.

If these multi-graph characters are to be encoded, they should be given canonical equivalences to resolve the issue of multiple equivalent representations. The decomposition mappings required are listed in Table 2:

Character	Decomposition mapping
1B06 BALINESE LETTER AKARA TEDUNG	1B05 1B35
1B08 BALINESE LETTER IKARA TEDUNG	1B07 1B35
1B0A BALINESE LETTER UKARA TEDUNG	1B09 1B35
1B0C BALINESE LETTER RA REPA TEDUNG	1B0B 1B35
1B12 BALINESE LETTER OKARA TEDUNG	1B11 1B35
1B3B BALINESE VOWEL SIGN RA REPA TEDUNG	1B3A 1B35
1B3D BALINESE VOWEL SIGN LA LENGA TEDUNG	1B3C 1B35
1B40 BALINESE VOWEL SIGN TALING TEDUNG	1B3E 1B35
1B41 BALINESE VOWEL SIGN TALING REPA TEDUNG	1B3F 1B35
1B43 BALINESE VOWEL SIGN PEPET TEDUNG	1B42 1B35

Table 2. Decomposition mappings for multi-graph vowel characters

Accordingly, the core Unicode character properties listed in L2/05-008 for these characters should be revised as follows:

```
1B06;BALINESE LETTER AKARA TEDUNG;Lo;0;L;1B05 1B35;;;;N;;aa;;;
1B08;BALINESE LETTER IKARA TEDUNG;Lo;0;L;1B07 1B35;;;;N;;ii;;;
1B0A;BALINESE LETTER UKARA TEDUNG;Lo;0;L;1B09 1B35;;;;N;;uu;;;
1B0C;BALINESE LETTER RA REPA TEDUNG;Lo;0;L;1B0B 1B35;;;;N;;vocalic rr;;;
1B12;BALINESE LETTER OKARA TEDUNG;Lo;0;L;1B11 1B35;;;;N;;au;;;
1B3B;BALINESE VOWEL SIGN RA REPA TEDUNG;Mc;0;NSM;1B3A 1B35;;;;N;;vocalic rr;;;
1B3D;BALINESE VOWEL SIGN LA LENGA TEDUNG;Mc;0;NSM;1B3C 1B35;;;;N;;vocalic ll;;;
1B40;BALINESE VOWEL SIGN TALING TEDUNG;Me;0;NSM;1B3E 1B35;;;;N;;o;;;
1B41;BALINESE VOWEL SIGN TALING REPA TEDUNG;Me;0;NSM;1B3F 1B35;;;;N;;au;;;
1B43;BALINESE VOWEL SIGN PEPET TEDUNG;Mc;0;L;1B42 1B35;;;;N;;;;;
```

It is worth pointing out that, among the characters proposed, there are several cases in which a character resembles another character or character sequence but is not equivalent. These are listed in Table 3:

Character	Similar character sequence	Comment
1B02 BALINESE SIGN CECEK	1B64	1B64 is a symbol and has a larger, spacing glyph
1B03 BALINESE SIGN SURANG	1B65	1B65 is a symbol and has a larger, spacing glyph
1B04 BALINESE SIGN BISAH	1B6A	1B6A is a symbol and has a spacing glyph
1B0D BALINESE LETTER LA LENGA	1B52	1B52 is a digit
1B0F BALINESE LETTER EKARA	1B56	1B56 is a digit
1B11 BALINESE LETTER OKARA	1B53	1B53 is a digit
1B28 BALINESE LETTER PA KAPAL	1B58	1B58 is a digit
1B35 BALINESE VOWEL SIGN TEDUNG	1B61	1B61 is a symbol and has a spacing glyph
1B36 BALINESE VOWEL SIGN ULU	1B66	1B66 is a symbol and has a larger, spacing glyph
1B38 BALINESE VOWEL SIGN SUKU	1B63	1B63 is a symbol and has a larger, spacing glyph
1B39 BALINESE VOWEL SIGN SUKU ILUT	1B68	1B68 is a symbol and has a larger, spacing glyph
1B3C BALINESE VOWEL SIGN LA LENGA	1B44 1B2E 1B42	
1B3E BALINESE VOWEL SIGN TALING	1B62	1B62 is a symbol and has a spacing glyph
1B42 BALINESE VOWEL SIGN PEPET	1B67	1B67 is a symbol and has a larger, spacing glyph
1B50 BALINESE DIGIT ZERO	1B5C	1B5C is punctuation with no numeric value
1B52 BALINESE DIGIT TWO	1B0D	1B0D is a word-forming letter with no numeric value
1B53 BALINESE DIGIT THREE	1B11	1B11 is a word-forming letter with no numeric value
1B54 BALINESE DIGIT FOUR	1B60	1B60 is punctuation with no numeric value
1B56 BALINESE DIGIT SIX	1B0F	1B0F is a word-forming letter with no numeric value
1B58 BALINESE DIGIT EIGHT	1B28	1B28 is a word-forming letter with no numeric value
1B5C BALINESE WINDU	1B50	1B50 is a digit
1B60 BALINESE PAMENENG	1B54	1B54 is a digit
1B61 BALINESE MUSICAL SYMBOL DONG	1B35	1B35 is a word-forming combining mark with a smaller glyph
1B62 BALINESE MUSICAL SYMBOL DENG	1B3E	1B3E is a word-forming combining mark
1B63 BALINESE MUSICAL SYMBOL DUNG	1B38	1B38 is a word-forming combining mark with a smaller glyph
1B64 BALINESE MUSICAL SYMBOL DANG	1B02	1B02 is a word-forming combining mark with a smaller glyph
1B65 BALINESE MUSICAL SYMBOL DANG SURANG	1B03	1B03 is a word-forming combining mark with a smaller glyph
1B66 BALINESE MUSICAL SYMBOL DING	1B36	1B36 is a word-forming combining mark with a smaller glyph
1B67 BALINESE MUSICAL SYMBOL DAENG	1B42	1B42 is a word-forming combining mark with a smaller glyph
1B68 BALINESE MUSICAL SYMBOL DEUNG	1B39	1B39 is a word-forming combining mark with a smaller glyph
1B6A BALINESE MUSICAL SYMBOL DANG GEDE	1B04	1B04 is a word-forming combining mark

Table 3. Similar but distinct character sequences

As mentioned in L2/05-008, similarities with other existing, non-Balinese characters may also exist.

Canonical combining classes

The purpose of canonical combining classes is to establish appropriate equivalence classes under Unicode normalizations for character sequences that involve combining marks. Specifically:

- Given a pair of combining marks that interact typographically (i.e., that nominally occupy the same position relative to the base), different encoded orders correspond to visually-distinct relative positions of the marks, hence are semantically distinct. By assigning these marks to the *same* canonical combining class (zero or non-zero), the *non*-equivalence of differently-ordered sequences is established under normalization.
- Given a pair of combining marks that *do not* interact typographically (i.e., that occupy *distinct* positions relative to the base), different encoded orders are visually identical, hence not semantically distinct. By assigning these marks to *different, non-zero* canonical combining classes, the *equivalence* of differently-ordered sequences is established under normalization.

Balinese text can contain combining sequences consisting of multiple combining marks. These can include multiple combining marks that *do* interact typographically; e.g. < 1B36 VOWEL SIGN ULU, 1B03 SIGN SURANG > (both occur above the base character). Or they can include multiple combining marks that *do not* interact typographically; e.g. < 1B38 VOWEL SIGN SUKU, 1B03 SIGN SURANG > (one occurs above the base, while the other occurs below). Therefore, it is relevant to consider what canonical combining classes would be appropriate for Balinese combining marks.

In L2/05-008, all combining marks are assigned to class 0, with two exceptions:

- 1B34 BALINESE SIGN REREKAN, which corresponds to the nukta in various South Asian scripts, is assigned to class 7, which is used for nuktas.
- 1B44 BALINESE ADEG ADEG, which is a virama, is assigned to class 9, which is used for viramas.

All other combining marks are assigned to class 0, however, and class 0 has special behaviour in the Unicode normalization algorithms: if a sequence contains a combining mark in class 0 and a mark in a non-zero class *n*, equivalence classes are defined as though the class-0 mark belonged to class *n*; i.e., that sequence is not equivalent to the sequence containing those marks in the opposite order. For instance,

< 1B38 VOWEL SIGN SUKU (ccc=0), 1B34 SIGN REREKAN (ccc=7) >

≢ < 1B34 SIGN REREKAN (ccc=7), 1B38 VOWEL SIGN SUKU (ccc=0) >

Using the canonical combining classes proposed in L2/05-008, there is only one pair of combining marks for which distinct orders would be considered canonically equivalent:

< 1B34 SIGN REREKAN (ccc=7), 1B44 ADEG ADEG (ccc=9) >

≡ < 1B44 ADEG ADEG (ccc=9), 1B34 SIGN REREKAN (ccc=7) >

Note that REREKAN and ADEG ADEG do not interact typographically (REREKAN is above while ADEG ADEG is to the right); hence this particular result is appropriate.

Several Balinese combining marks occur above the base character:

Character	Class
1B00 BALINESE SIGN ULU RICEM	0
1B01 BALINESE SIGN ULU CANDRA	0
1B02 BALINESE SIGN CECEK	0
1B03 BALINESE SIGN SURANG	0
1B34 BALINESE SIGN REREKAN	7
1B36 BALINESE VOWEL SIGN ULU	0
1B37 BALINESE VOWEL SIGN ULU SARI	0
1B42 BALINESE VOWEL SIGN PEPET	0
1B6B BALINESE MUSICAL SYMBOL COMBINING TEGEH	0
1B6D BALINESE MUSICAL SYMBOL COMBINING KEMPUL	0
1B6E BALINESE MUSICAL SYMBOL COMBINING KEMPLI	0
1B6F BALINESE MUSICAL SYMBOL COMBINING JEGOGAN	0
1B70 BALINESE MUSICAL SYMBOL COMBINING KEMPUL WITH JEGOGAN	0
1B71 BALINESE MUSICAL SYMBOL COMBINING KEMPLI WITH JEGOGAN	0
1B72 BALINESE MUSICAL SYMBOL COMBINING BENDE	0
1B73 BALINESE MUSICAL SYMBOL COMBINING GONG	0

Table 4. Balinese above-base combining marks

Combinations of syllable-modifier signs (1B00—1B03), REREKAN and vowel signs, at least, are linguistically valid. Because all of these but REREKAN are assigned to class 0, differently-ordered sequences of these marks, which would be visually distinct, are not canonically equivalent. Thus, the use of class 0 provides appropriate results in these cases.

Smaller numbers of Balinese combining marks position below the base character or to the left of the base character:

Character	Class
1B38 BALINESE VOWEL SIGN SUKU	0
1B39 BALINESE VOWEL SIGN SUKU ILUT	0
1B3A BALINESE VOWEL SIGN RA REPA	0
1B6C BALINESE MUSICAL SYMBOL COMBINING ENDEP	0

Table 5. Balinese below-base combining marks

Character	Class
1B3E BALINESE VOWEL SIGN TALING	0
1B6F BALINESE VOWEL SIGN TALING REPA	0

Table 6. Balinese left-of-base combining marks

Because of the meanings they signify, combinations of below-base marks and combinations of left-of-base marks should not occur. Also, for the below-base marks, with the possible exception of 1B6C BALINESE MUSICAL SYMBOL COMBINING ENDEP, there is no obvious way to combing these visually. In principle, though, one can assume that different encoded orders of some combination of below-base marks or of left-of-base marks could correspond to different visual results, for which purpose the use of class 0 is adequate. In practice, some implementations may prevent combinations of below marks or combinations of left-of-base marks, or treat such combinations as invalid sequences should they occur.

The sets of Balinese marks that position to the right of the base are similarly small:

Character	Class
1B04 BALINESE SIGN BISAH	0
1B35 BALINESE VOWEL SIGN TEDUNG	0
1B44 BALINESE ADEG ADEG	9

Table 7. Balinese right-of-base combining marks

In this case, < 1B35 BALINESE VOWEL SIGN TEDUNG, 1B04 BALINESE SIGN BISAH > is a linguistically plausible combination (though I do not know if it is ever actually used). Assuming it's normal use as a vowel killer, ADEG ADEG should not co-occur with either of the other two marks. Again, though, different encoded orders of a combination of these marks are possible in principle and would be visually distinct, and so the use of class 0 provides appropriate results in these cases.

In the cases described above, the use of class 0 is sufficient to cause differently-ordered combinations of marks that *do* interact typographically (having different visual results) to be considered *not* canonically equivalent. Where assignment of marks to class 0 breaks down, however, is in failing to cause differently-ordered combinations of marks that *do not* interact typographically to be considered *canonically equivalent*. Thus, in each of the following examples, a given visual text element has multiple encoded representations that are non-canonically-equivalent:

< 1B13 KA, 1B42 VOWEL SIGN PEPET, 1B04 SIGN BISAH >
≠ < 1B13 KA, 1B04 SIGN BISAH, 1B42 VOWEL SIGN PEPET >

< 1B13 KA, 1B34 SIGN REREKAN, 1B39 VOWEL SIGN SUKU ILUT >
≠ < 1B13 KA, 1B39 VOWEL SIGN SUKU ILUT, 1B34 SIGN REREKAN >

< 1B13 KA, 1B3E VOWEL SIGN TALING, 1B03 SIGN SURANG >
≠ < 1B13 KA, 1B03 SIGN SURANG, 1B3E VOWEL SIGN TALING >

< 1B13 KA, 1B39 VOWEL SIGN SUKU ILUT, 1B04 SIGN BISAH >
≠ < 1B13 KA, 1B04 SIGN BISAH, 1B39 VOWEL SIGN SUKU ILUT >

Of course, numerous other examples could also be supplied.

This failing in the canonical combining classes could be overcome if non-zero classes were used. Alternate assignments following the standard classes associated with various positions relative to the base are shown in Table 8:

Character	Class[3]
◌ᭀ 1B00 BALINESE SIGN ULU RICEM	230
◌ᭁ 1B01 BALINESE SIGN ULU CANDRA	230
◌ᭂ 1B02 BALINESE SIGN CECEK	230
◌ᭃ 1B03 BALINESE SIGN SURANG	230
◌᭄ 1B04 BALINESE SIGN BISAH	226
◌ᭅ 1B34 BALINESE SIGN REREKAN	7
◌ᭆ 1B35 BALINESE VOWEL SIGN TEDUNG	226
◌ᭇ 1B36 BALINESE VOWEL SIGN ULU	230
◌ᭈ 1B37 BALINESE VOWEL SIGN ULU SARI	230
◌ᭉ 1B38 BALINESE VOWEL SIGN SUKU	220
◌ᭊ 1B39 BALINESE VOWEL SIGN SUKU ILUT	220
◌ᭋ 1B3A BALINESE VOWEL SIGN RA REPA	220
◌᭎ 1B3E BALINESE VOWEL SIGN TALING	224
◌᭏ 1B3F BALINESE VOWEL SIGN TALING REPA	224
◌᭐ 1B42 BALINESE VOWEL SIGN PEPET	230
◌᭑ 1B44 BALINESE ADEG ADEG	9
◌᭫ 1B6B BALINESE MUSICAL SYMBOL COMBINING TEGEH	230
◌᭬ 1B6C BALINESE MUSICAL SYMBOL COMBINING ENDEP	220
◌᭭ 1B6D BALINESE MUSICAL SYMBOL COMBINING KEMPUL	230
◌᭮ 1B6E BALINESE MUSICAL SYMBOL COMBINING KEMPLI	230
◌᭯ 1B6F BALINESE MUSICAL SYMBOL COMBINING JEGOGAN	230
◌᭰ 1B70 BALINESE MUSICAL SYMBOL COMBINING KEMPUL WITH JEGOGAN	230
◌᭱ 1B71 BALINESE MUSICAL SYMBOL COMBINING KEMPLI WITH JEGOGAN	230
◌᭲ 1B72 BALINESE MUSICAL SYMBOL COMBINING BENDE	230
◌᭳ 1B73 BALINESE MUSICAL SYMBOL COMBINING GONG	230

Table 8. Alternate canonical combining classes for Balinese combining marks

[3] If it is assumed that 1B34 REREKAN (nukta) always modifies the base directly, creating a new base letter, and so remains in a fixed position directly above the base without changing its position relative to other above-base marks, then it can be assigned to fixed-position class 7. Otherwise, if it can re-order relative to other above-base marks, it would be in class 230. Similarly, if 1B44 ADEG ADEG is assumed to remain in a fixed position relative to the base, never re-ordering with other right-of-base marks (e.g., always immediately next to the base), then it can be assigned to fixed-position class 9. Otherwise, if it can re-order relative to other right-of-base marks, it would be in class 226.

Note, though, that the multi-part vowel marks such as 1B3B BALINESE VOWEL SIGN RA REPA TEDUNG, if encoded, cannot be assigned to standard position-based non-zero classes. Because the multi-part vowels involve components in two or more positions relative to the base, they interact with marks that would belong to two or more positional classes. For instance, VOWEL SIGN RA REPA TEDUNG interacts with marks that would belong to class 220 and simultaneously with marks that would belong to class 226. The multi-part vowel marks would need to behave as though they belong to two or more classes, yet there is no mechanism by which marks can be assigned to more than one class, and no single class exists that captures their multi-part nature.

The normalization algorithms always apply decomposition mappings before canonical combining classes are considered, however. As a result, as long as a multi-part vowel decomposes to simplex vowel marks, the canonical combining classes to which the multi-part vowel mark is assigned has no significance. Thus, non-zero classes could be used for Balinese vowel marks as long as all of the multi-part vowels decompose to sequences of simplex vowel marks. For that to be possible, one additional character would need to be encoded, corresponding to the lower component of 1B3C BALINESE VOWEL SIGN LA LENGA and 1B3D BALINESE VOWEL SIGN LA LENGA TEDUNG, and a decomposition mapping for 1B3C would have to be added:

Character	Decomposition mapping	Comment
1Bxx BALINESE VOWEL SIGN LA (??)		new character, below-base component of la lenga
1B3C BALINESE VOWEL SIGN LA LENGA	1B42 1Bxx	decomposition to sequence involving new character

Table 9. Additional character and decomposition mapping required to use non-zero classes

A choice must be made, therefore:

- encode multi-part vowels and assign all vowel marks to class 0, with the result that there will be non-canonically-equivalent representations that are visually identical;
- encode multi-part vowels, and encode new character corresponding to below-base component of LA LENGA; assign all marks other than multi-part vowels to standard, position-based non-zero classes; in so doing, avoid having multiple non-canonically-equivalent representations that are visually identical; or
- do not encode multi-part vowels; assign all marks to standard, position-based non-zero classes, thereby avoiding having multiple non-canonically-equivalent representations that are visually identical.

There are precedents for encoding the multi-part vowels and assigning all vowel marks to class 0 from several Indic scripts, such as Kannada. If this is done, then the issue of multiple representations must be dealt with by establishing user conventions that one representation is accepted for representing a given text element while all other representations with the same visual display are considered invalid or treated as mis-spellings, as shown in the following examples:

ᬓᭀ	< 1B13 KA, 1B40 VOWEL SIGN TALING TEDUNG> ≡ < 1B13 KA, 1B3E VOWEL SIGN TALING, 1B35 VOWEL SIGN TEDUNG >	*accepted representations*
ᬓᬵᬾ	< 1B13 KA, 1B35 VOWEL SIGN TEDUNG, 1B3E VOWEL SIGN TALING >	*invalid representation*

If the multi-part vowels are encoded and assigned to class 0, along with other simplex vowel marks, then a second choice must be made as to whether to use class 0 or non-zero classes for *other*, non-vowel, marks.

Representation of words that display *ra repa* but sort like *ra* + *pepet*

On page 3 of L2/05-008, in the section on ordering, the proposers describe an anomalous situation involving words that display one letterform but sort as though a different letterform were used. Briefly, a historic phonological distinction existed between *ra repa* versus *ra* + *pepet*, but this distinction was lost. This has led some users to change the spelling of certain words, using *ra repa* rather than the traditional spelling using *ra* + *pepet*. In sorting, however, these users have continued to sort these words as they had been under the traditional spelling. Thus, one dictionary sorts words beginning with *ra repa* in two distinct locations, as shown in Figure 1 and Figure 2:

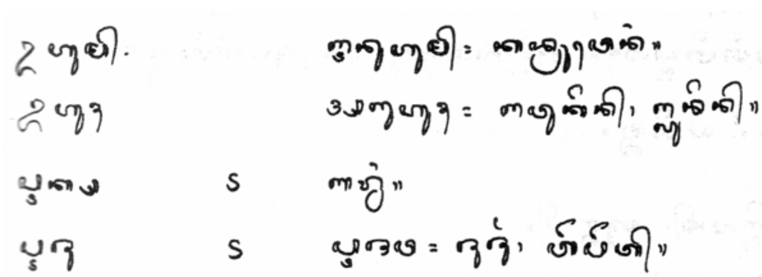

Figure 1. Words with *ra repa* sorting after *ukara* (Menaka 1990, p. 77)

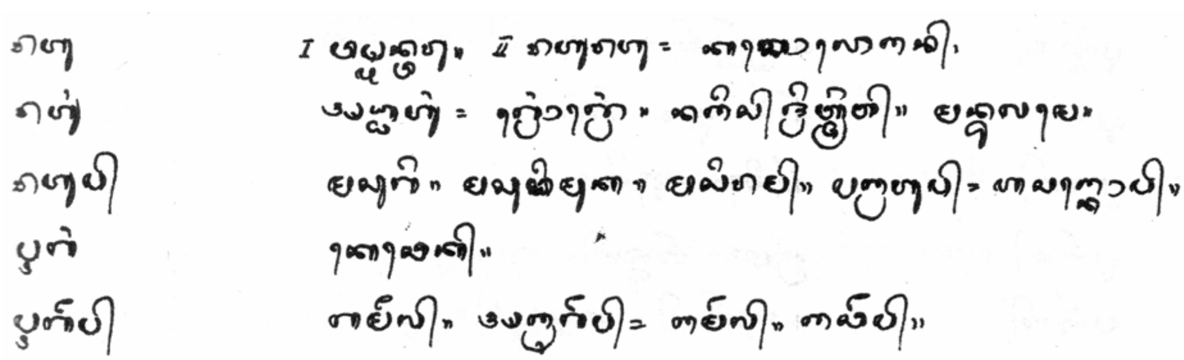

Figure 2. Words with *ra repa* sorting after *ra* (Menaka 1990, p. 347)

The proposers write (p. 3): "In effect, because the two sounds fell together, an orthographic congress at some point decided that words in *ᬭᭂ should always be written ᬋ. In order to account

for this anachronistic behaviour, fonts should render RA + PEPET as ᬋ, though an option to override this rendering should be made available to represent *ᬭᭂ."

The recommendation in the proposal to display the sequence < RA, PEPET > the same as < RA REPA > is problematic, in several respects:

- It results in multiple encoded representations for the same text element, which will lead to confusion for users and inconsistently-encoded data.
- It results in multiple letters being displayed like a single letter, which will lead to user confusion as well as confusion and complexity for implementers.
- It provides no direct representation for the traditional spelling in plain text. It relies instead on some unspecified rendering "override".
- It amounts to encoding the text in terms of historic phonemes rather than letterforms. It would be similar to encoding English text using U+00FE LATIN SMALL LETTER THORN but displaying that character as "th".
- Rather than treating the circumstance as a quirk in transitional, orthographic practice which, because of its quirkiness, will likely be short-lived, it imposes quirkiness on font and rendering implementations, which will likely be difficult to change.

Note that use of ZWJ or ZWNJ as the "override" mechanism for the proposed rendering would not be well-advised since those characters must already serve specific, well-defined functions for this Indic, virama-model script.

This recommendation in the proposal should be rejected. Rather than looking for an ad hoc solution within fonts and rendering sub-systems, it should be handled using mechanisms appropriate to orthographic / sorting anomalies. In particular, the alternate sorting for RA REPA could be accommodated by encoding sequences using CGJ, a character specifically intended to deal with alternate behaviours for processes such as sorting.