L2/24-058R

Required conjunct forms in extended grapheme clusters

Norbert Lindenberg, 2024-03-18

This document supersedes L2/23-141 and L2/24-058.

Proposal

This document proposes to change the definition of the property Indic_Conjunct_Break in UAX 44 Unicode Character Database, which defines values that are used in UAX 29 Unicode Text Segmentation in preventing extended grapheme cluster breaks within conjuncts and conjunct forms of some Brahmic scripts, as follows:

- Define the set of scripts that Indic_Conjunct_Break (InCB) applies to as the union of two sets, one for scripts whose conjunct forms are created using a character with Indic syllabic category Virama, the other for scripts that use a conjoiner with Indic syllabic category Invisible_Stacker. All scripts included in Unicode 15.1 go into the first set.
    - SV = [\p{sc=Beng}\p{sc=Deva}\p{sc=Gujr}\p{sc=Mlym}\p{sc=Orya}\p{sc=Telu}]
      SIS = []
      S = [SV||SIS]
- Add Balinese and Javanese to the first set:
    - SV = [\p{sc=Bali}\p{sc=Beng}\p{sc=Deva}\p{sc=Gujr}\p{sc=Java}\p{sc=Mlym}\p{sc=Orya}\p{sc=Telu}]
- Add Chakma, Dives Akuru, Kawi, Kharoshthi, Khmer, Meetei Mayek, Myanmar, Soyombo, Sundanese, Tai Tham (Lanna), Tulu-Tigalari, and Zanabazar Square to the second set:
    - SIS = [\p{sc=Cakm}\p{sc=Diak}\p{sc=Kawi}\p{sc=Khar}\p{sc=Khmr}\p{sc=Lana}\p{sc=Mtei}\p{sc=Mymr}\p{sc=Soyo}\p{sc=Sund}\p{sc=Tutg}\p{sc=Zanb}]
- Extend the definition of the InCB value Linker to allow all types of Unicode viramas that can create conjunct forms:

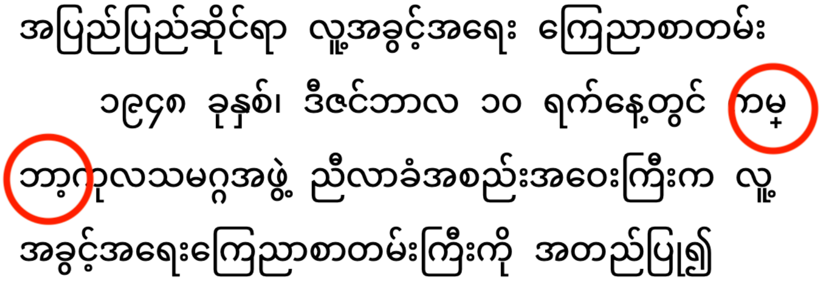
  - InCB = Linker iff C in [S&&[\p{Indic_Syllabic_Category=Virama}||\p{Indic_Syllabic_Category=Invisible_Stacker}]]
- Extend the definition of the InCB value Consonant to include all characters with Indic syllabic category Vowel_Independent of applicable scripts that use a conjoiner with Indic syllabic category Invisible_Stacker. Further extend it to include the Balinese characters ᬋ and ᬌ, which are known to have conjunct forms but do not have the Indic syllabic category Consonant:
  - InCB = Consonant iff C in [[S&&\p{Indic_Syllabic_Category=Consonant}]||[SIS&&\p{Indic_Syllabic_Category=Vowel_Independent}]||[\u{1B0B}\u{1B0C}]]
- Exclude the characters that now have InCB values Linker and Consonant from the set of characters having InCB=Extend. A proposal that handles this nicely is expected in the section "Error in the derivation of InCB" of L2/24-064 "UTC #179 properties feedback & recommendations" and is therefore omitted here.

Problems to be solved

Users of Brahmic scripts currently see a variety of problems where orthographic syllables are broken up in inappropriate ways. I do not know how the software products mentioned below are actually implemented, but in all cases the erroneous behavior is one that would result from the definition of extended grapheme clusters in UAX 29 combined with other Unicode Standard specifications.

Bad line breaks in Myanmar text

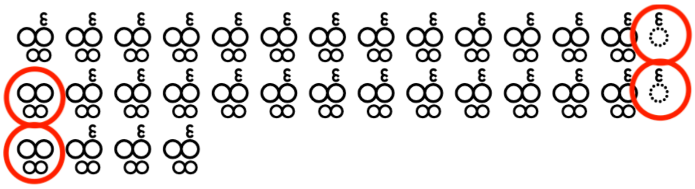
The following line break can be seen in Myanmar text (from the Burmese version of the Universal Declaration of Human Rights) in Apple's Safari browser:



အပြည်ပြည်ဆိုင်ရာ လူ့အခွင့်အရေး ကြေညာစာတမ်း
၁၉၄၈ ခုနှစ်၊ ဒီဇင်ဘာလ ၁၀ ရက်နေ့တွင် ကမ္
ဘာ့ကုလသမဂ္ဂအဖွဲ့ ညီလာခံအစည်းအဝေးကြီးက လူ့
အခွင့်အရေးကြေညာစာတမ်းကြီးကို အတည်ပြု၍

The syllable မ္ဘာ is broken at an extended grapheme cluster boundary, as is recommended by UAX 14 Unicode Line Breaking Algorithm as an emergency line break. This break is bad, as the “◌္” is not meant to ever be shown to users; it’s an “invisible stacker” that Unicode invented. (It is not clear why Safari uses an emergency line break here.)

The following line break can be seen in Myanmar test data in both Safari and the Chromium-based Brave browser:



Here the syllable င်္ထ္ထ is broken at an extended grapheme cluster boundary. This break is bad, as here the repha-like kinzi “◌င်္” is separated from the base on top of which it’s supposed to sit.

Bad backspace behavior in Balinese

When deleting text from the end of the Balinese greeting ᬒᬁᬲ᭄ᬯᬲ᭄ᬢ᭄ᬬᬲ᭄ᬢᬸ in Apple’s Pages app, the user sees successively:

ᬒᬁᬲ᭄ᬯᬲ᭄ᬢ᭄ᬬᬲ᭄ᬢᬸ
ᬒᬁᬲ᭄ᬯᬲ᭄ᬢ᭄ᬬᬲ᭄
ᬒᬁᬲ᭄ᬯᬲ᭄ᬢ᭄ᬬ
ᬒᬁᬲ᭄ᬯᬲ᭄ᬢ᭄
ᬒᬁᬲ᭄ᬯᬲ᭄
ᬒᬁᬲ᭄ᬯ
ᬒᬁᬲ᭄
ᬒᬁ

While this isn’t quite as bad as the Myanmar line breaking above, as the Balinese virama “◌᭄” is at least known to users, it is still rather surprising behavior. However, it is the result of deleting extended grapheme clusters, which is one of the two behaviors recommended in UAX 29 (the other alternative, deleting one code point at a time, is no better).

Required conjunct forms

This proposal addresses conjunct forms that are "required" in the sense that rendering the underlying character sequence as a conjunct form is the standard behavior, depending neither on context nor on choices made by a type designer. For most of the scripts covered by this proposal, the Unicode Standard invented a special character whose only purpose is be used in the representation of conjunct forms. Such characters have Indic syllabic category Invisible_Stacker, which implies that they should never be shown by themselves. In Balinese and Javanese, the Unicode Standard uses a character that can either combine with the following character to a conjunct form or remain visible; such characters have Indic syllabic category Virama.

Users often perceive conjunct forms themselves as base-level characters, and have names for them, such as *coeng ta* for the conjunct form ◌្ត of ត *ta* in Khmer, or *gantungan ka* for the conjunct form ◌᭄ᬓ of ᬓ *ka* in Balinese. The following table compares the user-perceived characters with extended grapheme clusters according to the current specification and the proposed version. Note in particular the "◌្" in Khmer, a character that Unicode invented and that does not exist in normal written Khmer.

	Khmer	Balinese
Text	ស្ត្រី	ᬲᬢ᭄ᬓ
User-perceived base-level characters	ស ◌្ត ◌្រ ◌ី	ᬲ ᬢ ◌᭄ᬓ
Extended grapheme clusters today	ស្ ត្ រី	ᬲ ᬢ᭄ ᬓ
Extended grapheme clusters proposed	ស្ត្រី	ᬲ ᬢ᭄ᬓ

Conjunct forms in Brahmic scripts are generally derived from consonants. In some scripts, however, conjunct forms also exist for some characters classified as independent vowels, especially vocalic liquids. Several cases are documented in the Unicode Standard: Section 16.4 Khmer shows conjunct forms for U+17A7 ឧ, U+17AB ឫ, U+17AC ឬ, and U+17AF ឯ; Section 17.3 Balinese for U+1B0B ᬋ (and U+1B0C ᬌ is canonical equivalent to <ᬋ, ◌ᬵ>); Section 17.9 Kawi for U+11F0A 𑼊 and U+11F0C 𑼌. The precise set for which conjunct forms exist is not always known. As characters of Indic syllabic category Invisible_Stacker are only used to create conjunct forms, using such a conjoiner with an independent vowel that doesn't have a conjunct form is invalid. We can therefore include all independent vowels in scripts that use an Invisible_Stacker (including Kawi and Khmer) into the set of "consonants". For scripts that use a character with Indic syllabic category Virama this is not possible. For Balinese, for example, the character sequence ᭄ᬅ (virama followed by independent vowel ᬅ) is known to occur without creating a conjunct form (for examples see leaves 7A and 13B of Atlas Bhumi). For these scripts, this proposal therefore includes only those characters for which conjunct forms are documented in the Unicode Standard; it is likely that the set will grow over time.

For this proposal only required conjunct forms are considered where the virama-like character combines with a subsequent character (as opposed to a preceding one) to create a conjunct form, as for these combinations the current definition of extended grapheme clusters breaks the conjunct forms.

The virama-like characters, and thus scripts, covered in this proposal were selected as follows:

- For characters with InSC=Invisible_Stacker, it was assumed that conjunct form creation is always required, as these characters are not meant to be displayed by themselves. If such a character occurs without a character with which it can combine, the text is incorrect, and the result of segmentation may be arbitrary. The only question is whether the character must combine with a subsequent character (as opposed to only with a preceding character). The block descriptions in the Unicode Standard or script proposals make clear that this always the case for the scripts proposed, except for the Myanmar *kinzi*, which is discussed below. In the Masaram Gondi and Gunjala Gondi scripts, the virama produces a half form of the preceding consonant, so it is not required that it be kept together with the subsequent consonant.
- For characters with InSC=Virama, a script-by-script analysis is necessary to determine whether conjunct form creation is required, contextual, or discretionary, whether the virama combines with subsequent or only with preceding characters, and which role zero width joiners might play. The proposal therefore includes only the two scripts with which the author is sufficiently familiar to perform this analysis, Balinese and Javanese. Experts in other scripts are invited to submit their own proposals.

A Myanmar *kinzi* is a special conjunct form that is encoded as a three-character sequence <Consonant, Pure_Killer, Invisible_Stacker> before the consonant on top of which it should be displayed. In this case the Invisible_Stacker combines with two preceding characters. However, it should not be separated from the base consonant on top of which it is displayed, so preventing grapheme cluster breaks between the *kinzi* and the subsequent consonant is still correct.

Compatibility with UAX 29 rule GB9c

In Unicode 15.1, a new rule GB9c was added to the list of rules that identify extended grapheme cluster breaks. This rule was earlier proposed in L2/18-147 as

> Virama [[Extend-\p{ccc=0}] ZWJ]* × LinkingConsonant

In the PAG recommendations that proposed the rule for Unicode 15.1, L2/23-079, this became

> LinkingConsonant ExtCccZwj* Virama ExtCccZwj* × LinkingConsonant

Additional changes were made to the rule before Unicode 15.1 became final; those changes are not relevant to the discussion here.

The most important change between the rules proposed in L2/18-147 and L2/23-079 is the addition of a requirement that the Virama-LinkingConsonant sequence to be kept together must be preceded by another LinkingConsonant, separated only by a limited set of extension characters. To date, no explanation has been provided why the context of a Virama-LinkingConsonant sequence should matter, and I believe this change was a mistake.

What makes this change particularly bad, however, are its assumptions that the sets of consonants that can occur before or after a virama in a syllable with conjunct forms are equal, and that the sets of permissible extending characters before and after the virama are equal. These assumptions have no basis in reality, and cause problems for several of the scripts included in this proposal:

- In the Balinese script, when used for Sasak, the vowel ᬅ can have other consonants subjoined (Unicode Standard section 17.3), but, as noted above, it does not have a conjunct form. In rule GB9c, it should therefore be part of the first set of consonants, but not the second.
- In the Khmer script, when used for Middle Khmer, conjunct forms can represent final consonants and then be encoded after robat, consonant shifters, ZWJ, ZWNJ, vowels, and other signs (Unicode Standard section 16.4). Since the Unicode Standard and shaping engines allow this encoding order, it is also sometimes erroneously used for modern Khmer. Thus the first set of extending characters should contain all these characters. Applying the rule constraint to "ignore degenerates" (UAX 29 section 1.2), most of these characters could be poured into the second set of extending characters as well, but one of them must not: ZWNJ is used in a number of scripts to prevent a virama and consonant from forming a conjunct form, and then should also cause a grapheme cluster break.

- For Tai Tham, there's no agreement yet on a standard encoding order, but it may face similar issues as Khmer, as it also represents final consonants with conjunct forms.

Fixing rule GB9c is beyond the scope of this proposal. The proposed changes eliminate a large number of undesirable breaks in extended grapheme clusters. With a future version of rule GB9c that drops context checking the problems described above would also be fully addressed. If a future version of rule GB9c instead just separates the sets of consonants and extending characters before and after the virama, more characters could be added to the pre-virama sets to handle the problems described. In the meantime, conjunct forms attached to Balinese ᬅ or used as final consonants in Khmer would be split into separate grapheme clusters, like they are today.

Feedback on earlier proposals

L2/23-141 discusses feedback on earlier proposals to fix extended grapheme clusters. None of the feedback should prevent this proposal from moving forward.

᭛᭜᭛