Date: August 2, 2010

L2/10-281

To: Unicode Technical Committee

From: Peter Edberg (Apple Inc.)

Subject: Fix to Unicode 5.1 changes for Grapheme_Cluster_Break Extend value

In Unicode 5.1, UAX #29 UNICODE TEXT SEGMENTATION added the notion of ***extended grapheme clusters*** versus ***legacy grapheme clusters,*** and updated the Grapheme Cluster Boundary Rules and the Grapheme_Cluster_Break property values in an attempt to support both cluster types. Unfortunately, the updated property values do not allow the intended implementation of legacy grapheme clusters as described in UAX #29, though legacy grapheme clusters are described in UAX #29 as if the rules and values do support their implementation.

To add extended grapheme clusters, the original rule GB9 was left alone:

GB9		×	Extend

and new rules were added for extended grapheme clusters:

GB9a.		×	SpacingMark
GB9b.	Prepend	×	

Unfortunately, the Extend class was also changed to include characters appropriate for extended grapheme clusters but not appropriate (in most cases) for legacy grapheme clusters, by adding the following Thai and Lao characters (mostly spacing postfix vowels):

U+0E30 (ะ) THAI CHARACTER SARA A
U+0E32 (า) THAI CHARACTER SARA AA
U+0E33 (ำ) THAI CHARACTER SARA AM
U+0E45 (ๅ) THAI CHARACTER LAKKHANGYAO
U+0EB0 (ະ) LAO VOWEL SIGN A
U+0EB2 (າ) LAO VOWEL SIGN AA
U+0EB3 (ຳ) LAO VOWEL SIGN AM

Apple and CLDR have both received feedback that Thai users much prefer legacy-style grapheme clusters for most text editing operations such as cursor movement and character deletion, as well as for default alignment of search strings. UAX #29 should correctly support the implementation of legacy grapheme clusters. However, Thai users have also indicated that the postfix vowel U+0E33 THAI CHARACTER SARA AM should be included in a legacy-style grapheme cluster.

We have not had direct feedback concerning Lao, though the same consideration is likely to apply, with U+0EB3 LAO VOWEL SIGN AM preferably included with legacy grapheme clusters.

Therefore we propose that the Grapheme_Cluster_Break property values for the following be changed from Extend to SpacingMark: U+0E30, U+0E32, U+0E45, U+0EB0, and U+0EB2. The Extend and SpacingMark sections in UAX #29 Table 2 should thus be changed as follows (the shaded characters are moved from Extend to SpacingMark):

Extend	Grapheme_Extend = true, or U+0E33 (◌ำ) THAI CHARACTER SARA AM U+0EB3 (◌ຳ) LAO VOWEL SIGN AM This includes: General_Category = Nonspacing_Mark General_Category = Enclosing_Mark U+200C ZERO WIDTH NON-JOINER U+200D ZERO WIDTH JOINER plus a few Spacing Marks needed for canonical equivalence.
Prepend	…
SpacingMark	General_Category = Spacing Mark *and* Grapheme_Cluster_Break ≠ Extend, or U+0E30 (◌ะ) THAI CHARACTER SARA A U+0E32 (◌า) THAI CHARACTER SARA AA U+0E45 (◌ๅ) THAI CHARACTER LAKKHANGYAO U+0EB0 (◌ະ) LAO VOWEL SIGN A U+0EB2 (◌າ) LAO VOWEL SIGN AA

This will not change the behavior of extended grapheme clusters, but will restore the behavior of legacy grapheme clusters (with the enhancement for SARA AM / AM).

The following annexes copy relevant material from the Unicode 5.0 version and subsequent versions of UAX #29, for reference.

Annex A: UAX #29 for Unicode 5.0 (tr29-11.html); selected sections.

…

A default grapheme cluster begins with a base character, except when a nonspacing mark is at the start of text, or when it is preceded by a control or format character. In place of a single base character, a Hangul syllable composed of one or more characters may serve as the base. For the rules defining the default boundaries, see Table 2. For more information on the composition of Hangul syllables, see Chapter 3, Conformance, of [Unicode].

…

Table 2. Grapheme_Cluster_Break Property Values

Extend	Grapheme_Extend = true

…

Grapheme Cluster Boundary Rules

…

Do not break before extending characters.

GB9 × Extend

…

Annex B: UAX #29 for Unicode 5.1 (tr29-13.html) through current draft for Unicode 6.0 (tr29-16.html); the sections copied below have not changed across those versions.

…

Table 1a. Sample Grapheme Clusters

Extended grapheme clusters		
நி	U+0BA8 (ந) tamil letter na U+0BBF (ி) tamil vowel sign i	Tamil ni
เก	U+0E40 (เ) thai character sara e U+0E01 (ก) thai character ko kai	Thai ke
षि	U+0937 (ष) devanagari letter ssa U+093F (ि) devanagari vowel sign i	Devanagari ssi

A ***legacy grapheme cluster*** is defined as a base (such as A or 力) followed by zero or more continuing characters. One way to think of this is as a sequence of characters that form a "stack".

The base can be single characters, or be any sequence of Hangul Jamo characters that form a Hangul Syllable, as defined by D118 in The Unicode Standard.

The continuing characters include nonspacing marks, plus the Join Controls (U+200C ( ) ZERO WIDTH NON-JOINER and U+200D ( ) ZERO WIDTH JOINER used in Indic languages, and a few spacing combining marks to ensure canonical equivalence. Additional cases need to be added for completeness, so that any string of text can be divided up into a sequence of grapheme clusters. Some of these may be *degenerate* cases, such as a control code, or an isolated combining mark.

An ***extended grapheme cluster*** is the same as a legacy grapheme cluster, with the addition of some other characters. The continuing characters are extended to include all spacing combining marks, such as the spacing (but dependent) vowel signs in Indic scripts, as continuing characters. For example, this includes U+093F (ि) DEVANAGARI VOWEL SIGN I.

The definition also includes certain visual order Thai and Lao vowels that may come before the base. The extended grapheme clusters should be used in implementations in preference to legacy grapheme clusters, because they provide better results for Indic scripts such as Tamil or Devanagari, and for Southeast Asian scripts such as Thai and Lao.

…

Table 1b. Combining character sequences and grapheme clusters

legacy grapheme cluster	(CRLF | (Hangul-syllable | !Control) Grapheme_Extend* | .)	A single base character is a grapheme cluster. Degenerate cases include any isolated non-base characters, and non-base characters like controls.
extended grapheme cluster	(CRLF | **Prepend*** (Hangul-syllable | !Control) (Grapheme_Extend **| Spacing_Mark)*** | .)	Extended grapheme clusters add prepending and spacing marks

…

Table 2. Grapheme_Cluster_Break Property Values

…

Extend	Grapheme_Extend = true, or U+0E30 (ะ) THAI CHARACTER SARA A U+0E32 (า) THAI CHARACTER SARA AA U+0E33 (ำ) THAI CHARACTER SARA AM U+0E45 (ๅ) THAI CHARACTER LAKKHANGYAO U+0EB0 (ະ) LAO VOWEL SIGN A U+0EB2 (າ) LAO VOWEL SIGN AA U+0EB3 (ຳ) LAO VOWEL SIGN AM This includes: General_Category = Nonspacing_Mark General_Category = Enclosing_Mark U+200C ZERO WIDTH NON-JOINER U+200D ZERO WIDTH JOINER plus a few Spacing Marks needed for canonical equivalence
Prepend	…
SpacingMark	General_Category = Spacing Mark *and* Grapheme_Cluster_Break ≠ Extend

…

Grapheme Cluster Boundary Rules

…

Do not break before extending characters.

GB9. × Extend

Only for extended grapheme clusters:
Do not break before SpacingMarks, or after Prepend characters.

GB9a.		×	SpacingMark
GB9b.	Prepend	×	

…