L2/13-004

Proposal to change General Category of MONGOLIAN VOWEL SEPARATOR from Zs to Cf

Behdad Esfahbod
Google
December 20, 2012

Proposed Action

In UTC 133, it was approved to add Mongolian and ‘Phags-pa to ArabicJoining.txt. I like to propose to make the following changes to properties of MVS:

1. Change General_Category of MVS from “Zs” to “Cf”, hence making it Default_Ignorable again,
2. Change Joining_Type of MVS from implicit U to explicit U.

Background

In UTC 133 I brought up a proposal to make MONGOLIAN VOWEL SEPARATOR a Default_Ignorable codepoint. At the meeting, it was noted that above mentioned character used to be Default_Ignorable in the past and such status was changed in UTC 93. The committee (rightly) decided to not take any action until further research is done.

The particular change was discussed in UTC 93 as part of L2/02-368 "Default Ignorable Issues", which sought to clean up the definition of Default_Ignorable and the characters matching that property. I like to bring to the committee’s attention the following paragraph from L2/03-368:

> Default-ignorable code points are those that should be ignored by default in rendering (unless explicitly supported). They have no visible glyph or advance width in and of themselves, although they may affect the display, positioning, or adornment of adjacent or surrounding characters. Some of the default ignorable code points are assigned characters, while others are reserved for future assignment.

The following consensus was reached at UTC 93:

> **[93-C10] Consensus**: For Unicode 4.0, change the general category of U+180E MONGOLIAN VOWEL SEPARATOR from "Cf" to "Zs"; add it to the "whitespace" property; remove from list of Default Ignorables; and ensure that the line break property is no-break, in order to align it with narrow no-break space. [L2/02-368]

and as such this character was recategorized as Whitespace, and hence not Default_Ignorable, on the basis that it results in a visual space where it occurs.

Discussion

I like to argue that the consensus reached at UTC 93 is one approach to categorizing this character ("like a narrow no-break space"), but there is room for alternate approaches. I would like to suggests that this character may more closely reflect properties of ZERO WIDTH NON-JOINER, which is much closer in its function to the MONGOLIAN VOWEL SEPARATOR.

To quote Peter Constable from UTC 133 (my recollection of his words): "There seems to be a narrow space by virtue of glyphs not joining." Which, I hope we all agree, is also the case for ZERO WIDTH NON-JOINER used in all the scripts with Arabic-like joining behavior (including Mongolian). And ten years after that discussion in UTC 93, at least the Microsoft and HarfBuzz implementations of Mongolian do exactly that as far as I can see: treat MVS as a NonJoining character, and not render it like Default_Ignorable characters.

Now, lets compare the pros and cons of the two different approaches in implementations that do and do not support this particular character.

	"Supporting" implementation	"Non-supporting" implementation
Current properties	Correct	Correct joining, spurious "missing" glyph shown
Proposed properties	Correct	Correct joining, no spurious glyphs shown

One reason to move MVS away from the Whitespace category is that it simply isn't one. When a font designer is designing a font, for each character in the Whitespace category, then can read the character properties and design a glyph with a certain advance width for it. For MVS it just does not work that way. Not in any implementations ten years after the change AFAIK.

I would also like to suggest a more nuanced model for defining Default_Ignorable. As L2/02-368 discussed in detail, pretty much all characters in Default_Ignorable affect some behavior of their surrounding context (line breaking, joining, bidi, etc), but what they also have in common is that when it comes to rendering, they don't have a shape or advance width in and of themselves. This is an interesting model because it allows separating layout and rendering of Unicode text into a number of separable stages (bidi, line-breaking, script-specific behavior, choosing glyphs, etc), whereas each stage can have its own "support" status for different sets of characters. And then Default_Ignorable can simply be defined as "choose no glyphs for these", while at the same time other stages can "support" behavior specific to a Default_Ignorable character.

Summary

I think the decision made in UTC 93 was a valid approach to addressing the issues under discussion, but when it comes to MVS, that's not the only possible approach. I would like to suggest the proposed action to be considered by the UTC, which, in my opinion, will provide better fallback behavior for implementations not supporting this particular character, but also making it easier to implement this character for implementations that choose to do so, and make the standard more consistent by categorizing MVS and ZWNJ more closely.