L2/16-208

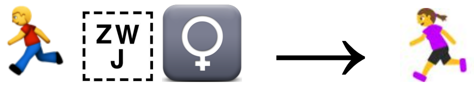
Re: Future-proofing emoji ZWJ sequences
From: Mark Davis, Peter Edberg, and the CLDR committee
Live: https://goo.gl/cluFCn
Data: GlueAfterZwj.txt
Ticket:#9438

This document provides background for CLDR ticket #9438, which adds a property and rule changes in CLDR 3.0 to future-proof emoji zwj sequences, so that possible future emoji zwj sequences will not break grapheme clusters, words, or lines.

For Unicode 9.0, the minimal changes were made to segmentation (UAX #29 and UAX #14) for the current ZWJ sequences. Although it was expected that many more ZWJ sequences will be encoded, it was too late in the development cycle for Unicode 9.0 for it to be adjusted to account for that. Instead, it was agreed that the next version of CLDR should provide customizations of segmentation that would work. The UTC motion was 147-M1.

> **[147-M1] Motion:** In view of the timeline for Unicode 9.0 changes proposed in L2/16-093 would be better implemented in the near term as customizations in CLDR or ICU.

The following shows an example of a such new sequence, using a character that would break lines according to stock Unicode 9.0.



1F3C3 200D 2640

Importantly, for users the bad effects from breaking too much around ZWJ are *far worse* than those from breaking too little around ZWJ. So the rules can err on the side of being overly inclusive when it comes to symbols. The worst that happens is that a pictograph or emoji symbol “glues” to a previous letter in linebreak, but that is a minor issue for users.

The following table illustrates this.

- Cell A1 shows the problem case, where a ZWJ sequence of a female runner breaks after the ZWJ. So part of the sequence shows at the end of one line, and the rest on the next line.
- Cell B1 shows that this is fixed with the customizations. Either the whole zwj sequence fits on the line or it is wrapped to the next.
- Cells A3 shows the side effect; a sequence consisting of a letter + ZWJ + symbol is treated as a whole for line breaking. Such sequences are, however, extremely rare in practice and the behavior is not that bad — the only effect is that the sequence may wrap together to the next line.

	Source strings	A. No customizations	B. Customizations
1	4 *women running*	🏃‍♀️ 🏃‍♀️ 🏃‍♀️ 🏃 ♀️	🏃‍♀️ 🏃‍♀️ 🏃‍♀️ 🏃‍♀️
2	3 *women running* + *“o”* + *ZWJ* + *female sign*	🏃‍♀️ 🏃‍♀️ 🏃‍♀️o ♀️	🏃‍♀️ 🏃‍♀️ 🏃‍♀️ o♀️

Customizations

The customizations consist of a binary character property, and customizations of segmentation rules in UAX #29 and UAX #14. The property is Glue_After_Zwj (GAZ), with values in GlueAfterZwj.txt.

Feedback on these values is welcome for CLDR v30, and can be added as comments to Ticket #9438, or by filing a new CLDR ticket.

The property data includes pictographs, characters that are related to current emoji, and some ranges of unassigned code points (Cn). The Cn characters based on the Line_Break assignments that were added in Unicode 9.0 to help future-proof ZWJ linebreak behavior even for future characters (not just new usage of existing characters in emoji zwj sequences).

The rule customizations are as follows:

BASE RULES (Unicode v9.0)

GB11	ZWJ ×	(Glue_After_Zwj | EBG)
WB3c	ZWJ ×	(Glue_After_Zwj | EBG)
LB8a	ZWJ ×	(ID | EB | EM)

Key for base rules:

E_Base	**EB**	Characters which are bases for emoji modifiers (Emoji_Modifier_Base=Yes)
E_Modifier	**EM**	Characters which are emoji modifiers (Emoji_Modifer=Yes)
Glue_After_Zwj	**GAZ**	Characters which can occur in emoji sequences after ZWJ, but which are not also bases for emoji modifiers (Emoji_Modifier_Base=No)
EBase_GAZ	**EBG**	Characters which can occur in emoji sequences after ZWJ, and are also bases for emoji modifiers (Emoji_Modifier_Base=Yes)

CUSTOMIZED RULES (Unicode CLDR v30)

Let **Glue_After_Zwj** be redefined as in GlueAfterZwj.txt
(overriding the GB and WB property values!)
Let **Emoji** = \p{Emoji=Yes}

GB11′	ZWJ ×	(Glue_After_Zwj | Emoji)
WB3c′	ZWJ ×	(Glue_After_Zwj | Emoji)
LB8a′	ZWJ ×	(ID | Glue_After_Zwj | Emoji)

Review Notes

The ID value is retained in LB8a′. The main reason for its inclusion was to encompass as many Emoji characters as possible, and was not otherwise needed. So it could be removed from LB8a′. However, it is retained here to allow for as much compatibility with Unicode 9.0 as possible — it does little harm to keep it.

We could define Glue_After_Zwj to be the data in the file plus \p{Emoji=Yes}, which would make the statement of the customizations functionally clearer.

References

- L2/16-093
- L2/16-094
- L2/16-181