L2/24-057R

Title: Proposal of the EAST ASIAN AUTO SPACING
Authors: Koji Ishii (Google), Yasuo Kida (W3C), Fuqiao Xue (W3C)
Date: Jul 1, 2024
Supersedes: L2/24-057, L2/23-283. See the Modifications.

EAST ASIAN AUTO SPACING (Proposal)

1 Overview and Scope

East Asian text usually consists of multiple scripts, such as Han ideographs, Kana syllables, and Hangul syllables, along with Latin letters and numeric characters. East Asian established typography conventions define that a thin spacing between East Asian scripts and other scripts improves the readability. This spacing should be represented by adjusting glyph spacings similar to kerning, rather than by using space characters.

While detailed rules of the spacing can vary across documents, it is important that the choice made by an author for a specific document be clearly established, so that a rendering system can display what the author intended. It is also important that this choice be established independently of the font resources, as the rendering systems may have to use other fonts than those intended or specified in the document. Finally, the expression of the author's choice should be relatively concise, to facilitate document authoring and minimize document size.

This report describes a Unicode character property which can serve as a stable default rule of inserting the spacing for the purpose of reliable document interchange.

For the purpose of reliable document interchange, this property defines an unambiguous default value, so that implementations could reliably render a character stream based solely on the property values, without depending on other information such as provided in the tables of the selected font.

The intent is that the author should be able to specify where they want to override, and that in the absence of an explicit specification, the spacing is implicitly that defined by the property presented in this report.

The actual choice for the property values should result in a reasonable or legible default, but it may not be publishing-material quality, and it may not be a good choice if used in a specific style or context.

The property values are chosen first to match existing practice in East Asian contexts in their respective environments. For characters that are not generally used in such environments, similarity to existing characters has been taken into consideration. It also takes East Asian characters in non-East Asian texts into account.

2 Conformance

The property defined in this report is informative. The intent of this report is to provide, in the absence of other information, a reasonable way to determine the correct automatic spacing, but this behavior can be overridden by inserting space characters, or by a higher-level protocol, such as through markup or by the preferences of a layout application. This default determination is defined in the accompanying data file Data, but in no way implies that the spacing is inserted only by this rule.

For more information on the conformance implications, see Unicode, Section 3.5, Properties, in particular the definition (D35) of an informative property.

3 The Auto_Spacing Property (as)

3.1 Property Values

The possible property values are given in Table 1.

Table 1. Property Values

Value	Short Name	Description	Examples
W	Wide	Characters that are considered as East Asian script for the auto-spacing purpose.	Han ideographic characters and Kana syllables are examples of this value.
N	Narrow	Characters that need the auto-spacing with adjacent “W” characters.	Latin letters and digits are examples of this value.
O	Other	Characters that don’t need the auto-spacing.	Most symbols and punctuation characters such as COMMA and FULL STOP are examples of this value.
C	Conditional	Characters that are “N” for Chinese and “O” for other languages.	Characters that can appear as prefix or suffix of Latin letters

or digits, such as U+0025 PERCENT SIGN.

Characters that have the property value “N” are similar to the “Narrow” characters in UAX#11 EAST ASIAN WIDTH, but most punctuation characters and symbols are excluded. Similarly, characters that have the property value “W” are similar to the “Wide” characters, but most punctuation characters and symbols are excluded. Also, to follow the existing practice, Hangul characters, circled characters, square characters, and Emoji are defined as “O”.

For the value “C”, please refer to 3.2.3 Symbols and Punctuation Characters for more details.

3.2 Spacing Algorithm

The auto-spacing should be inserted between “W” and “N”, and between “N” and “W”, after resolving the conditional value “C” to “N” or “O”.

The exact amount of the spacing can vary across documents. This property doesn’t define the exact amount. Instead, it should be defined by high-level protocols or applications such as through markup or by the preferences of a layout application.

There are two ways to represent a space: a character space (by the insertion of physical code points), or in a glyph space (similar to kerning, adjusting the metrics of adjacent glyphs on the device). High-level protocols or applications should use glyph spaces where possible.

3.3 Scope of the Property

3.3.1 Grapheme Cluster

As in all matters of typography, the interesting unit of text is not a character, but a grapheme cluster: it does not make sense to insert the auto spacing between a base character and a combining mark attached to it. Implementations should insert the auto-spacing before or after each grapheme cluster.

A possible choice for the notion of grapheme cluster is either that of legacy grapheme cluster or that of extended grapheme cluster, as defined in UAX#29.

The property value for a grapheme cluster as a whole is then determined by taking the property value of the first character in the cluster, with the following exception:

- If the cluster contains an enclosing combining mark (general category Me), then the whole cluster has the Auto_Spacing property value “O”.

3.3.2 Space Characters

The property values for space characters (General Category Zs) are “O”. This is to avoid inserting the auto-spacing around space characters, which can lead to undesirable double spacing.

It also allows authors to override the algorithm when high-level protocols or applications don’t provide a way for authors to express their intent to override this algorithm, such as plain text files.

U+0020 SPACE indicates a semantic boundary, which is stronger than the spacing for the readability. Using the code point for the auto-spacing purpose can make distinguishing semantic boundaries from the spacing for the readability difficult, and therefore it’s discouraged.

U+2009 THIN SPACE should usually represent a thin space, which is suitable to represent the auto-spacing for the readability. Inserting the code point to where the algorithm doesn’t insert the auto-spacing should indicate that the auto-spacing is desired there.

Likewise, inserting U+200B ZERO WIDTH SPACE to where the algorithm inserts the auto-spacing should prevent the auto-spacing from being inserted by rendering systems.

3.3.3 Symbols and Punctuation Characters

In some existing practices, symbols and punctuation characters insert the spacing, while they don’t in other existing practices.

For example, when one side of a word is a letter or a digit and the other side is a punctuation character, such as “20%”, “$20”, or “C#”, and they appear adjacent to “W” characters, some existing practices prefer inserting the spacing to both sides of them, considering that not doing so look unbalanced.

On the other hand, some other existing practices prefer not inserting the spacing between punctuation characters and “W” characters even in such cases. They view the spacing as a way to secure legibility by preventing East Asian letters from being too close to other letters and numeral digits, rather than to highlight words as parentheses do.

The conditional value “C” is assigned to such characters. Chinese content often prefers the spacing in such cases, and thus they should resolve “C” to “N”, while other content should resolve “C” to “O”.

If the author is uncertain whether their content is used in Chinese context or not, and they want to express their intentions, they can override the algorithm as described in 3.3.2 Space Characters.

3.3.4 Vertical Text Layout

In vertical text layout, a character may be displayed upright or sideways rotated, as defined in UAX#50.

If a character that has the Auto_Spacing property value “N” is displayed upright, the rendering system should handle it as if it has the property value “O” instead.

3.3.5 Right-to-Left Scripts

This property has a current limitation in that the handling of right-to-left scripts is not specified. This includes scripts that are predominantly written right to left, such as Arabic, along with right-to-left scripts that are meant to be written vertically, such as Chorasmian.

4 Data File

This property is derived by the following algorithm:

1. Assign the property value “W” if it’s in the following set:
    1. Include if the Script property is one of the following values: Bopomofo (Bopo), Han (Hani), Hangul (Hang), Hiragana (Hira), Katakana (Kana), Khitan_Small_Script (Kits), Nushu (Nshu), Tangut (Tang), Yi (Yiii).
    2. Include if the Script_Extensions property is one of the values above, except when the East_Asian_Width property is “N” or “Na”.
    3. Exclude if the East_Asian_Width property is “H”.
    4. Exclude if the General_Category property is “P” or “No”.
    5. Exclude if the General_Category property is “S” except “Sk”.
    6. Include the following code point: U+3013 GETA MARK.
2. Otherwise, assign the property value “C” if it’s in the following set:
    - Include if the General_Category property is “Po”.
    - Exclude if the East_Asian_Width property is “F”, “H”, or “W”.
    - Exclude the following code points: U+0022 QUOTATION MARK, U+0027 APOSTROPHE, U+002A ASTERISK, U+002F SOLIDUS, U+00B7 MIDDLE DOT,

U+2020 DAGGER, U+2021 DOUBLE DAGGER, U+2026 HORIZONTAL ELLIPSIS.

3. Otherwise, assign the property value “N” if it’s in the following set:
    1. Include if the General_Category property is “L”, “M”, or “Nd”.
    2. Exclude if the East_Asian_Width property is “F, “H”, or “W”.
4. Otherwise, assign the property value “O”.

The derived data file and a python code are availble for references.

Modifications

Modifications from L2/24-057

- Renamed to "East Asian Auto Spacing" from "Unicode Auto Spacing".
- Added the property value “C”, and updated 3.3.3 Symbols and Punctuation Characters.
- Added short names to the property values.
- Changed Hangul and Yi to “W”.
- Fixed to use the correct code point U+2009 THIN SPACE for a thin space.
- Updated Data File to the up-to-date algorithm and links.
- Removed all open issues.
- Editorial updates to: 1 Overview, 3.2 Spacing Algorithm, 3.3.1 Grapheme Cluster, 3.3.2 Space Characters.