

DATE: 9 June 2018
DOC TYPE: Working Group Document
TITLE: Comments on WG2 #67 documents (June 2018)
SOURCE: Deborah Anderson, Ken Whistler, Roozbeh Pournader, Andrew Glass, Peter Constable, Lisa Moore, Marek Jeziorek, and Ben Yang
STATUS: Expert contribution
ACTION: For consideration by WG2
DISTRIBUTION: ISO/IEC JTC1/SC2/WG2

The following are comments by a group of experts on WG2 #67 documents

1. Khitan Small Script (N4943R=L2/18-121R)

We reviewed this document, which discussed the encoding model for Khitan Small Script (KSS), which is one of the controversies surrounding this script.

A. Encoding Model Discussion

There are two cluster patterns in Khitan Small Script. The prevalent pattern, “Cluster A”, starts with two adjacent KSS characters, and ends with either a single centered character or two adjacent KSS characters. The alternate cluster pattern (Cluster B type) starts with a single KSS character.

Cluster A type

2	3	4	5	6	7	8
①②	①② ③	①② ③④	①② ③④ ⑤	①② ③④ ⑤⑥	①② ③④ ⑤⑥ ⑦	①② ③④ ⑤⑥ ⑦⑧

Cluster B type

2	3	4
① ②	① ②③	① ②③ ④

Three options to handle KSS clustering are listed below.

1. CGJ (N4943R)

The following summarizes the approach advocated in N4943R (see table below, p. 3):

- The model uses a Combining Grapheme Joiner (CGJ) after the first character to indicate the Cluster B pattern. CGJ was chosen for its line-breaking properties (i.e., it prevents line-break before and after the CGJ). The author states that he is open to a KSS-specific format character “if the committees consider that would be more appropriate” (noted in the table below with G’).

- A space character (U+0020 or another space character) is used to separate clusters from one another. Clusters are terminated by EOL, CGJ, or any non-KSS graphic character.
- The cursor control would allow users to click into the middle of a Khitan cluster or to select a portion of the cluster. Any characters in the stack can be removed one at a time using forward delete and backspace depending on the position of the caret.
- The author has a working implementation of this model, using his prototype font.
- New OpenType features would be required in order to substitute the correct positional glyph form for each KSS character in a run.

Comments:

- Use of space to separate clusters would be a new model, adding complexity and creating a burden for any major vendor to implement, if the vendor wishes to mitigate the issues with the editing model.
- Editing is not ideal, since one can't see where the cursor is in a stack. Preferable would be an *aksara*-type cluster, in which caret placement cannot be inserted within the cluster, forward delete removes the entire stack and backspace removes one character at a time. This is the general model used for complex script clusters, e.g., Tibetan, Devanagari clusters. The closest existing model to the proposed one is the Old Hangul model. This requires dedicated features and a dedicated engine to implement.
- Model requires OpenType features that are not yet supported.

2. EGYPTIAN HIEROGLYPH MODEL ([N4768](#)=L2/16-338)

The WG2 meeting in September 2016 recommended a different solution, one based on the model that has been adopted for Egyptian hieroglyphs.

Summary of this approach (see table below, see p. 3):

- Two format characters—a horizontal joiner and a vertical joiner—are used between characters in a cluster.
- Has the advantage of already being implementable.
- No special OpenType features are required.
- Would require an input method so users don't need to type a joiner character between each character in the cluster.

3. GLUE CHARACTERS

A third alternative has been suggested by Andrew Glass.

Summary of the model (see table below, see p. 3):

- Two format characters are required: one "glue" character is inserted between each KSS character in the cluster. To indicate that a single KSS starts a cluster (Cluster B type), a second glue character is required.
- Would require an input method so users don't need to type a "glue" character between each character in the cluster.
- No special OpenType features are required.

Recommendations: Our recommendation is to choose an approach that fits within a recognized model and presents fewer barriers to implementation by vendors. Such an approach would be less ad hoc and will be able to guarantee wider implementation into the future. The Egyptian Hieroglyph model is preferred, but the “glue” alternative is also acceptable.

The patterns proposed would be as follows (where “K” designates a KSS character):

	Cluster A	Cluster B	
	K K	K	
	K K	K K	
		K	
			Notes
West:	sp. K K K K .sp	sp. K CGJ K K K .sp	.sp = space or start of run
West':	sp. K K K K .sp	sp. K G' K K K .sp	G' = Glue control char.
WG2 2016:	K * K : K * K	K : K * K : K	* = horizontal join : = vertical join
Glass:	K G K G K G K	K G' K G K G K	G and G' = Glue characters

B. Other recommendations:

Radicals

We agree that removal of the 12 non-clustering radicals from the KSS repertoire is warranted, since they “are not used in Khitan Small Script texts” (p. 2, N4795=[L2/17-107](#)).

Iteration Mark

We recommend the KHITAN SMALL SCRIPT ITERATION MARK be moved from the KSS block into the Ideographic Symbols and Punctuation block at U+16FE2, after the Tangut and Nushu iteration marks. This relocation will simplify chart production and prevent errors for implementers. Moving the iteration mark to the 16FE0 block follows the precedent set for Tangut and Nushu.

Currently, all KSS characters are named by code point (“KHITAN SMALL SCRIPT CHARACTER-18XXX”), except for the iteration mark. Chart production is made more complicated when a non-ideographic character – which has its own name and properties – is added into a block. In addition, inclusion of the iteration mark in the KSS block can lead to errors for implementers, since it will require a one-off property change inside the block, a change not usually expected for large ideographic scripts. Keeping the iteration mark in the KSS block will also cause problems for UnicodeData.txt, which would have to split the range for the ideographic part of the block and require a one-off line with the name for the iteration mark in the block. Lastly, a font mapping to include the iteration mark from another small block is trivial (cf. inclusion of characters from CJK Symbols and Punctuation block in the U+3000 block in CJK fonts).

Chart

Because of the back-and-forth regarding inclusion of radicals and location of the iteration mark, we







recommend a new chart be created without the 12 radicals (such as in N4795=L2/17-107), but move the iteration mark to U+16FE2 in the Ideographic Symbols and Punctuation block. (For a list of past KSS documents, see: <https://www.unicode.org/L2/topical/khitan/>.) The chart need only refer back to the main proposal (N4725R) for background details on the script.

2. Zanabazar Square (N4945)

Document: [L2/18-132](#) Proposal to encode two additional Zanabazar Square letters (WG2 N4945) – Andrew West

Comments: We reviewed this proposal for two Zanabazar Square cluster-initial letters LA and SA, which correspond to Tibetan head letters LA and SA that appear in conjuncts. The two proposed characters correspond to already encoded character U+11A3A ZANABAZAR SQUARE CLUSTER-INITIAL LETTER RA.

The author identifies a distinction between Sanskrit transcription and Tibetan transcriptions in Zanabazar Square, where the Sanskrit examples show the two consonant glyphs with a gap between the glyphs, and the Tibetan transcription depicts a compressed ligature, as shown in the chart on page 1:

	RA + KA	LA + KA	SA + KA
Sanskrit Syllable			
Tibetan Syllable			

The following comments arose during discussion:

- Clarification is needed on how to handle cases that fall outside the “typical” examples shown on page 1 (above). How should the following be encoded: Figure 3 (compressed and not ligated), Figure 5 (halfway compressed and not all clearly ligated), Figure 7 (halfway compressed and ligated), and Figure 11 (compressed and not ligated). If no clear guidelines are provided, encoding these two new characters might result in encoding LA and SA in two ways, with no way to distinguish them.
- Provide examples showing contrastive use in a single source.

It was noted that the proposal modified the glyph for the currently encoded U+11A3A CLUSTER-INITIAL LETTER RA, changing the dotted box to a dotted circle. The dotted box, indicating special rendering, appears in the Soyombo code chart for the four cluster-initial letters. We believe the dotted circle in place of the dotted box is an oversight of the author.