

Universal Multiple-Octet Coded Character Set  
International Organization for Standardization  
Organisation Internationale de Normalisation  
Международная организация по стандартизации

**Doc Type: Working Group Document****Title: Discussion of Cluster Formation in Khitan Small Script****Source: Andrew West, Michael Everson, Viacheslav Zaytsev****Status: Individual Contribution****Action: For consideration by JTC1/SC2/WG2 and UTC****Date: 2016-10-27****1. Background**

L2/16-113 (WG2 N4725) Section 4 suggested several different possible mechanisms for controlling cluster formation in the Khitan Small Script. L2/16-156 “Recommendations to UTC #147 May 2016 on Script Proposals” recommended the use of two Khitan-specific format characters: a horizontal and a vertical stacker. This recommendation was endorsed at the May UTC meeting, and we revised L2/16-113 (WG2 N4725) to reflect our understanding of this recommendation. (In retrospect, it may have been the case that our understanding of the recommended model was flawed.)

The model we presented to the Meeting on Khitan Scripts held at Yinchuan, China in August 2016 was for two format characters, KHITAN SMALL SCRIPT DOUBLE CLUSTER INITIAL that causes a sequence of characters to form a cluster starting with two adjacent characters, and KHITAN SMALL SCRIPT SINGLE CLUSTER INITIAL that causes a sequence of characters to form a cluster starting with a single character. This model was accepted by the experts at the Yinchuan meeting (see L2/16243 / WG2 N4736).

At WG2 Meeting 65 in San José the Ad Hoc Meeting on Khitan Small Script decided not to accept the model agreed at Yinchuan, but rather to use the encoding model described in section 9 of L2/16-156 with the use of two format characters, KHITAN SMALL SCRIPT HORIZONTAL JOINER and KHITAN SMALL SCRIPT VERTICAL JOINER (see WG2 N4768). However, L2/16-156 does not actually describe how these two format characters would be applied, so there is uncertainty as to how exactly cluster-formation would be controlled in Khitan Small Script. This document discusses possible problems with the encoding model using horizontal and vertical joiners (as we understand that it is intended to work), and suggests a modification to this model.

**2. Discussion of Model Accepted at WG2 M65**

For the purpose of the following discussion we posit the following two example clusters, with the letters A through E representing Khitan Small Script characters (see N4725R Tables 1 through 3 for examples of actual Khitan Small Script clusters):

<i>Cluster A</i>	<i>Cluster B</i>
<b>AB</b>	<b>A</b>
<b>CD</b>	<b>BC</b>
<b>E</b>	<b>DE</b>

As we understand it, either KHITAN SMALL SCRIPT HORIZONTAL JOINER (\*) or KHITAN SMALL SCRIPT VERTICAL JOINER (: ) would be placed before each character in a cluster other than

the first character (for any sequence of  $n$  Khitan characters forming a cluster,  $(n - 1)$  format characters would be required). So a sequence of five Khitan characters  $\langle ABCDE \rangle$  forming *Cluster A* would be represented with nine characters as  $\langle A*B:C*D:E \rangle$ , and a sequence of five Khitan characters  $\langle ABCDE \rangle$  forming *Cluster B* would be represented with nine characters as  $\langle A:B*C:D*E \rangle$ .

<i>Cluster A</i>	<i>Cluster B</i>
<b>A*B:</b>	<b>A:</b>
<b>C*D:</b>	<b>B*C:</b>
<b>E</b>	<b>D*E:</b>

Putting a joiner character between every pair of characters in a cluster is very flexible, and allows for the formation of clusters with arbitrary shapes. For example, the sequence  $\langle A*B*C:D*E:F:G*H:I*J*K \rangle$  would create an hourglass-shaped cluster of 3-2-1-2-3 characters.

*Cluster C*  
**A\*B\*C:**  
**D\*E:**  
**F:**  
**G\*H:**  
**I\*J\*K**

This flexibility is important for Egyptian hieroglyphs which may form arbitrary quadrats, but is *not* appropriate for Khitan Small Script, because Khitan Small Script clusters are not arbitrary but have a fixed format. The hourglass-shaped cluster posited above *cannot* occur in Khitan Small Script, and should be considered an illegal cluster shape.

All Khitan clusters consist of one or more levels, with one centred or two adjacent characters on the top and bottom levels, and two adjacent characters on any intervening levels. Thus, for any sequence of Khitan Small Script characters only two cluster shapes are possible, one starting with two adjacent characters (the normal case), and one starting with a single centred character (the unusual case). So for the sequence of five Khitan characters  $\langle ABCDE \rangle$  the only possible cluster shapes are *Cluster A* and *Cluster B* shown above. This means that once the placement of the first character in the cluster is known, the placement of the remaining characters in the cluster is fixed.





As the first format character determines the placement of the first character in the cluster, all the following format characters are redundant. Not only is more than one format character in a cluster redundant, but the flexibility of having a format character between each pair of characters in the cluster becomes a burden on both implementers and end users, because it allows for the formation of arbitrary illegal cluster shapes. Implementers would need to support such arbitrary cluster shapes, which would be considerably harder than just supporting legal cluster shapes. End users would need to cope with entering and manipulating a very high proportion of format characters in their text (a 90% increase for a five-character cluster from  $\langle ABCDE \rangle$  to  $\langle A*B:C*D:E \rangle$ ), and there would be a high possibility of format characters becoming lost or misplaced in the text.

### 3. Suggested Modification to Cluster-Forming Model

As only one format character is required to determine the cluster shape in Khitan Small Script, and having to deal with redundant format characters in each and every cluster in a text is an unnecessary burden to end users that would certainly inhibit the adoption of Unicode Khitan Small Script, we suggest that the cluster-forming model be modified to use only a single format character for each cluster, either KHITAN SMALL SCRIPT DOUBLE INITIAL CLUSTER MARKER (DICM) for a cluster starting with two adjacent characters or KHITAN SMALL SCRIPT SINGLE INITIAL CLUSTER MARKER (SICM) for a cluster starting with a single centred character. Moreover, in order to facilitate search operations for

sequences of characters that may be in either cluster shape, we suggest that the format character is placed in front of the first character in the cluster. With these modifications, *Cluster A* would be represented as <DICM ABCDE> and *Cluster B* would be represented as <SICM ABCDE>. This would allow for both sequences to be matched when searching for the string <ABCDE>, which is very important for corpus analysis.

We recommend that the format characters DICM and SICM are rendered as visible glyphs when not preceding a contiguous sequence of two or more Khitan Small Script characters, as this will aid in entering and editing Khitan Small Script text. The process for entering and displaying *Cluster A* and *Cluster B* using this model is shown below.

Step	Character inserted	Cluster A		Cluster B	
		Text stream	Result	Text stream	Result
1	DICM or SICM	<DICM>		<SICM>	
2	A	<DICM A>	 A	<SICM A>	 A
3	B	<DICM A B>	AB	<SICM A B>	A B
4	C	<DICM A B C>	AB C	<SICM A B C>	A BC
5	D	<DICM A B C D>	AB CD	<SICM A B C D>	A BC D
6	E	<DICM A B C D E>	AB CD E	<SICM A B C D E>	A BC DE

This is simple to implement and intuitive to the user. This solution would be preferable to inserting HJ or VJ between every two characters in every cluster, which would be burdensome to the end user type, to edit, and to process.