INCITS/L2/07- 079
Date: February 10, 2007

Title:	Comments accompanying the US negative vote on PDAM4 to ISO/IEC 10646:2003
Source:	INCITS/L2
Action:	Forward to INCITS

The US National body is voting No with comments on the following SC2 ballot. Satisfying technical comment T.1 would change the vote into a Yes.

SC2N3914: Information technology -- Universal Multiple-Octet Coded Character Set (UCS) -- AMENDMENT 4: Lanna, Cham, Game Tiles, CJK Unified Ideographs Extension C, and other characters

Technical Comments:

T.1 Character removal (Lanna)

The US is asking for the removal of the following characters:

```
1A65 LANNA VOWEL SIGN AM
1A66 LANNA VOWEL SIGN TALL AM
```

The rationale for their inclusion is provided in WG2 N3121:

> The presence of [LANNA VOWEL SIGN] AM (and [LANNA VOWEL SIGN] TALL AM) follows the Thai convention of ensuring that a final consonant is not stored before the vowel it follows. This is the only situation in which it could occur and so [LANNA VOWEL SIGN] AM is encoded to alleviate the problem.

It is again clarified in WG2 N3207:

> The written representation of /am/ involves two visual components: ◌ᩣ VOWEL SIGN AA (or ◌ᩤVOWEL SIGN TALL AA) and ◌ᩴ MAI KANG, which, if /am/ were not used, would be stored in that order (since final consonants are always stored after their vowels). In the case of /am/ the MAI KANG is often rendered as part of the preceding cluster to VOWEL SIGN AA. In order to ensure grapheme cluster integrity (see UAX#29 section 3) the unitary characters ◌ᩴᩣ and ◌ᩴᩤ for /am/ are proposed, following Thai practice. Note that /am/ is the only situation in which this occurs. The use of a sequence for AM would break the opportunity for a cluster boundary before AA. The characters may (if the UTC thinks it wise) be given compatibility decompositions to AA + MAI KANG and TALL AA + MAI KANG respectively. (In Thai, the decomposition for U+0E33 SARA AM is to 0E4D NIKHAHIT + U+0E32 SARA AA; this seems to be opposite, but Thai encodes in visual order so since the models are different this is not really relevant.)
>
> The AM characters are an example of how sometimes more than one solution can be proposed for an encoding problem. It could be argued that these are “duplicate” characters, though the compatibility decomposition mitigates against that. One of the chief problems is that Northern Thai treats AM similarly to Thai AM; it places the MAI KANG glyph to the left of the -AA vowel (whether over the previous cluster or between the clusters): ◌ᩴᩣ, ◌ᩴᩤ. In Khün and Lue, the MAI KANG render the MAI KANG over the -AA vowel: ◌ᩣᩴ, ◌ᩤᩴ. Without an encoded AM, it would be likely that Northern Thai users would confuse AA + MAI KANG and MAI KANG + AA, even though the latter is logically incorrect for the underlying phonemes. This is not a problem for Khün and Lue, which treat it as a vowel + final, but Northern Thai users think of it as equivalent to Thai AM.
>
> Potentially, MAI KANG and AA may also occur with MAI KANG properly preceding AA, in different syllables.

> For example /kam.wa:/ might be written ᨠᩴ᩠᩵ᩅᩤ= KAL + MAI KANG + TONE-1 + SAKOT + WA + TALL AA while /kwa:m/ would be written ᨠ᩠ᩅ᩵ᩤᩴ= KAL + SAKOT + WA + TONE-1 + TALL AM.
>
> The explicitly-encoded AM gets around the problems of the re-ordering and ligation that would have to be solved if there were no AM, and would add a complexity that is not present in any of the surrounding scripts that contribute to the encoding mileau [sic] of the intended user community.

However all that long and detailed explanation does not remove the fact that these two characters are in fact equivalent to sequences of characters which are also proposed for encoding in the same document.

<U+1A63, U+1A76> for LANNA VOWEL SIGN AM, and
<U+1A64, U+1A76> for LANNA VOWEL SIGN TALL AM.

Proposing compatibility decomposition makes them even less useful as they will be filtered out by all processes using normalization form KC. It also makes them unsuitable for identifiers where the alternate sequences would be the only allowed representation.

In all cases, duplicate encoding should not happen in new proposals.

T.2 Addition of 2 Lanna characters

The US is also supporting the addition of the following Lanna characters as proposed by document WG2 N3207:

```
1A29 LANNA LETTER KHUN HIGH CHA
1AAD LANNA SIGN CAANG
```

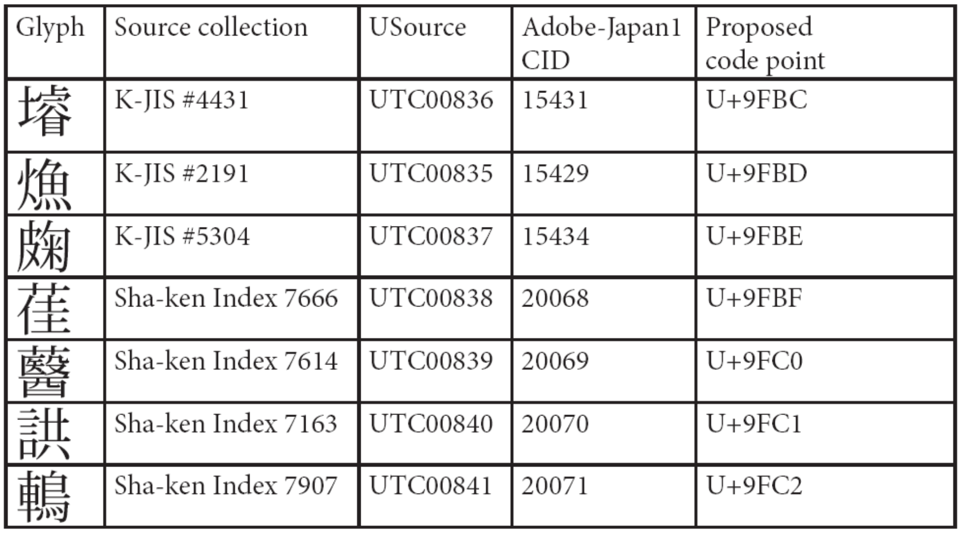
T.3 Name and glyph changes for the new Cyrillic Extended-A

The US is in favor of the glyph and name changes as proposed in WG2 N3194 for the characters in the range U+2DE0..U+2DF5 (code position as originally presented in document SC2 N3914).

T.4 Addition of 7 CJK Unified Ideographs

The US is also supporting the addition of 7 CJK Unified ideographs as proposed by document L2/07-67 (WG2 TBD) in positions U+9FBC through U+9FC2. At its last meeting, the IRG did not object to the fast-tracking of those characters, nor to their inclusion in Amendment 4. However, the IRG asked that those seven characters not be interleaved in Extension C, hence the proposed code points

These ideographs are present in the K-JIS and Sha-ken character collections. The K-JIS collection is developed by 共同通信社 and 配信先新聞社 for writing newspaper articles in Japan. The Sha-ken collection is part of a proprietary typesetting system widely used in Japan. These characters are also present in the Adobe-Japan1 collection, which is the basis for many desktop fonts, and at the time of this proposal are the only characters of that collection not present in Unicode / ISO/IEC 10646.



Glyph	Source collection	USource	Adobe-Japan1 CID	Proposed code point
龼	K-JIS #4431	UTC00836	15431	U+9FBC
龽	K-JIS #2191	UTC00835	15429	U+9FBD
龾	K-JIS #5304	UTC00837	15434	U+9FBE
龿	Sha-ken Index 7666	UTC00838	20068	U+9FBF
鿀	Sha-ken Index 7614	UTC00839	20069	U+9FC0
鿁	Sha-ken Index 7163	UTC00840	20070	U+9FC1
鿂	Sha-ken Index 7907	UTC00841	20071	U+9FC2

---End of US comments