L2/18-332

Tai Tham ad hoc meeting report

To: Unicode Technical Committee
From: Liang Hai <lianghai@gmail.com>
Date: 28 November 2018

A group of experts met in San Jose, California on 13 September 2018 for preliminary discussions about current issues of the Tai Tham encoding.

1. Participants

- Addison Phillips (Amazon and W3C)
- Andrew Glass (Microsoft)
- Chris Chapman (co-host, Adobe)
- Debbie Anderson (UC Berkeley and Unicode)
- Direk Injan / ดิเรก อินจันทร์ (Chiang Mai Rajabhat University / มหาวิทยาลัยราชภัฏเชียงใหม่) *
- Kamal Mansour (Monotype)
- Ken Lunde (co-host, Adobe)
- *Phra Bhikkhu* Khamchang Chantasaro / ᨻᩕᨽᩥᨠ᩠ᨡᩩᨤᩴᨧ᩠ᨦ ᨧᨶ᩠ᨴᩈᩣᩁᩮᩣ (Palm Leaf Academy of Sipsongpanna / 西双版纳贝叶书院) * **
- Liang Hai / 梁海 (Unicode)
- Lisa Moore (Unicode)
- Martin Hosken (SIL International) *
- Ned Holbrook (Apple) *
- Norbert Lindenberg (Lindenberg Software)
- Patrick Chew (Change.org)
- Pichai Saengboon / พิชัย แสงบุญ (Chiang Mai University / มหาวิทยาลัยเชียงใหม่) *
- Richard Ishida (W3C)
- Roozbeh Pournader / روزبه پورنادر (WhatsApp)
- *Phra Maha* Sangdang Arloka / ᨻᩕᨾᩉᩣᩈᩯ᩠ᨦᨯᩯ᩠ᨦ ᩋᩣᩃᩮᩣᨠ (Kengtung.org) * **
- Wang Yihua / 王奕桦 (Fudan University / 复旦大学) *

Notes in parentheses don’t imply formal representation.
* Participated remotely.
** Italicized part is monastic title.

2. Discussed topics

- Identities of the script and regional variants
- Encoding order restriction

- A visual-motivated encoding order
- Encoding order of consonant signs and vowel signs
- Poorly supported character sequence <C1 letter, V sign, C2 sign>
- Tone signs
- Khuen–Lue transferable OA BELOW
- MAI KANG LAI's various placement styles

3. Preliminary recommendations

- Existing script and language codes/tags in ISO 15924 and BCP 47 need to be stable, but documentation can be improved.
- Comparison and introduction of regional variants are valuable, but they should be recorded in dedicated documents instead of overloading the code chart.
- A restrictive encoding order (a cluster pattern) is helpful for minimizing text confusability and simplifying text processing.
- A typical visual order (Thai-style) is not necessary, but a visual-motivated encoding order is deemed helpful.
- A written structure <C1 base, C2 sign, V sign, C3 base> shall not be represented phonetically with <C1 letter, C2 sign, V sign, C3 sign>. (Note a consonant sign can be either an atomic character or a <stacker, consonant letter> sequence.)
- To represent a cluster, regardless of the phonetic order CCV or CVC, a consonant sign should always be encoded before the vowel sign, unless the vowel sign has inline advance and is apparently followed by the consonant sign.
- Vowel signs in a cluster should be encoded with the order "left, top, bottom, right".
- Shaping specifications (such as the Universal Shaping Engine) should be updated to allow a consonant sign to a follow a vowel sign.

4. Action items

- **A1. Patrick Chew:** Provide clarification and propose changes to ISO 15924 for the code "Lana".
- **A2. Patrick Chew:** Compare the regional variants Lanna, Khuen–Lue, and Lao–Isaan.
- **A3. The Tai Tham font developers:** Investigate the amount of existing Tai Tham data in Unicode, and see if this can be a barrier to modifying the encoding.
- **A4. Martin Hosken and Patrick Chew:** Investigate when a consonant sign (including both the regular and medial signs of LA) can be attached to a post-base vowel sign.
- **A5. Martin Hosken and Patrick Chew:** Confirm if syllable chaining or other mechanisms can lead to <C1 base, post-base V sign, below-base C2 sign (below V), above-base T sign (above V)>.
- **A6. Martin Hosken:** Document the different analyses of CVC and CCV for a syllable ending with -y.

- **A7. The native users:** Investigate in what order native users write the leading syllable's vowel sign OA BELOW and the following syllable's consonant letter when OA BELOW is transferred to the following syllable's base. (Arloka believes Khuen users write V first.)
- **A8. The Tai Tham font developers:** Investigate whether it is desired to have the multiple styles of placing MAI KANG LAI available as options in a single font, or it is acceptable to have them available only as a font-to-font variation.
- **A9. Martin Hosken and Patrick Chew:** Find out from the original proposal what the contrast is between MAI KANG LAI and the other sign in Khuen–Lue. Can they be unified in Khuen–Lue?
- **A10. Andrew Glass:** Investigate the progress of implementing REPHA in Microsoft's own USE implementation.
- **A11. Andrew Glass:** Provide exemplar fonts that demonstrate how to implement the multiple styles of placing MAI KANG LAI, and how to stop a shaping engine from reordering MAI KANG LAI when the reordering is not desired.

See Appendix B. *Meeting notes* for context of these action items.

5. Acknowledgements

The meeting was made possible with support from Adobe Inc. and the Unicode Consortium.

***

Appendix A. Topics carried over for future meetings

- **The visual-motivated encoding order's impact on phonetic sorting**
- **U+1A7B TAI THAM SIGN MAI SAM:** where should this combining mark be in the cluster pattern for the following functions of it?
  - A signal for the syllable boundary between a pair of stacked consonants. Alternatively it can be analyzed as a vowel restorer for the leading consonant.
  - A signal for a "double-acting consonant", which acts as both the leading syllable's final consonant and the following syllable's initial consonant.
  - A signal for that the whole syllable is repeated.
- **U+1A7A TAI THAM SIGN RA HAAM**

***

Appendix B. Meeting notes

B.1. Identities of the script and regional variants

- Can we change the ISO 15924 code "Lana" to something more appropriate?
  - Changing language codes always hurts communities. There would be strong pushback. Also, BCP 47's stability policy prevents changes.
    - *Ken Whistler's comment at 12 Oct's Script Ad Hoc:* If additional script tags are desired, see Syriac's model of variants (which are considered a single script in Unicode).
  - "(Lanna)" may be removed from the ISO 15924 code's script name. Can also consider subtags for the regional variants.
  - The Unicode Standard can carry notes and aliases. CLDR is generally open to altering script names and adding aliases.
  - **A1. Action item for Patrick Chew:** Provide clarification and propose changes to ISO 15924 for the code "Lana".
- Can the Unicode code chart include a comparison between regional variants, like what it does for CJK ideographs?
  - Difficult and expensive to maintain in the code chart.
  - Such guidance for implementers on regional variants should be illustrated in external documents (eg, a Unicode Technical Note). Unicode editors can then digest the infomation and see what should go into the *Core Specification*.
  - **A2. Action item for Patrick Chew:** Compare the regional variants Lanna, Khuen–Lue, and Lao–Isaan.
- Can a single font support both Lanna and Khuen–Lue variants?
  - Need to better understand the intended use cases of supporting both variants in a single font.
  - Different language tags or even different script tags may help.
  - Although unification is helpful for switching regional variants by simply switching the font, certain letters that have significant regional differences can be re-examined to see if disunification is necessary.
- Three major regional variants are identified: Lanna, Khuen–Lue, and Lao–Isaan.
  - Khuen and Lue variants are both culturally and structurally close to each other. Thus often when experts tend to only mention the Khuen variant, preferably the "Khuen–Lue" variant as a whole should be discussed.
  - Note there is a difference between letterform/structural variantion and stroke-style variation. The Lue variant superficially appears closer to Lanna because of its stroke style, but structurally it is closer to Khuen.
  - Note non-native experts sometimes over-emphasize the importance of differentiating regional variants, while native users are often used to a wide range of variants.
- "Tham" may be a more appropriate term than "Tai Tham".

B.2. Encoding order restriction

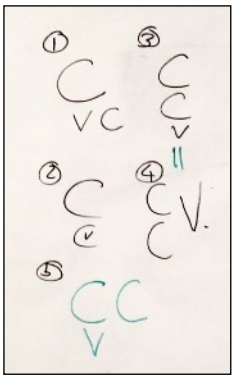
- **A3. Action item for the Tai Tham font developers:** Investigate the amount of existing Tai Tham data in Unicode, and see if this can be a barrier to modifying the encoding.
- Because the cluster structure is complex, the scope of a cluster should be discussed and clarified.
- The "encoding order" should not be understood as "typing order".
  - Aim for minimal confusability from the encoding order, and allow flexible orders in typing.
  - The most desirable encoding order and the most desirable typing order should be separated and resolved with different solutions.
  - Refer to how the issue of encoding order is dealt with for other scripts, such as Thai and Arabic.
  - We the industry owe users for not delivering encoding, fonts, and visual input solutions together.
- It is helpful if the agreed on cluster pattern does not diverge from the USE's current cluster pattern.
- USE's cluster validation is meant to restrict encoding order (not typing order) so text can be less ambiguous and easier to process. Eg, more consistent text is easier for searching.
  - In particular, USE restricts the encoding order of combining marks and inserts dotted circles to invalidly ordered marks because many Indic script combining marks have ccc = 0 (so confusable encoding orders are not fully normalized).
  - CVT+C is normalized to CV+TC because the canonical ordering process swaps the tone mark (ccc = 230, Above) and the stacker (ccc = 9, Virama). But this is not a problem for text engines, as long as acknowledged properly.
- The Unicode Standard does not actually specify the dotted circle mechanism. Instead, the *Core Specification* recommends an encoding order in the form of "use A; do not use B".
  - Implementations can then act accordingly. Different ways of rejecting bad encoding orders can exist at different levels of text processing.
- Can an additional normalization process be introduced to address ambiguity, allowing the encoding order to be less restrictive?
  - Unless such additional normalization is documented for all languages and is significantly helpful, we will not be able to force all vendors to implement it only for Tai Tham.

B.3. A visual-motivated encoding order

- A standard encoding order, visual-motivated but not visual-obsessed.
- An overly visual strategy can over-encourage encoding insignificant differences such as handwriting variation, and can make text processing such as searching more difficult.
- Ideally, users should be able to encode a written structure without knowing the pronunciation.
  - Encoding pronunciation means confusable ways of encoding, and is sometimes based on controversial pronunciations.
  - Move away from phonemics and focus on what users can see.

- Visual-motivated encoding impacts phonetic sorting.
  - Southeast Asian dictionaries usually are not sorted on written structures but on pronunciation. Lanna and Khuen dictionaries are often sorted as if words are converted to Thai. Less so for Lue.
  - Refer to similar cases: Japanese (pronunciation), Arabic and Persian (root).
  - Dictionaries can be phonetically sorted anyway.
  - Softwares can have pronunciation keys.
  - Make sure people are aware of the cost on sorting.
    - How much Unicode Tai Tham text exists?
- Existing fonts tend to be consistent in terms of expected encoding order, because current shaping engines are restrictive in the same way. But shaping engines will inevitably change in an effort to support Tai Tham properly.
- A major source of confusably encoded text is that languages utilize subjoined consonants for different syllable structures. A subjoined consonant is generally Sanskrit–Pali vs modern languages)
- Different numbers of stacking are allowed in different languages and regional variants.
- A written structure <C1 base, C2 sign, V sign, C3 base> shall not be represented phonetically with <C1 letter, C2 sign, V sign, C3 sign>.

B.4. Encoding order of consonant signs and vowel signs



- A consonant sign (C sign) can either be encoded as an atomic C sign (not necessarily medial) or a sequence <stacker, C letter>.
  - It is attested that a cluster can have two consonant signs.
  - The consonant signs' InSC values should be examined and the values might need to change if they're inappropriate.
- All the 5 cases (note the "C" on the right side of the case 5 was meant to be a post-base sign) of graphical structure that involve a base, a consonant sign, and a vowel sign should be encoded as <C1 letter, C2 sign, V sign>, regardless of the phonetic order.
- The left–top–bottom–right vowel sign order is good as the standard order when multiple vowel signs exist in a cluster.
  - Consistent with the USE.
- The "syllable chaining" mechanism allows a CVCV cluster to appear as either <C1 base, post-base V1 sign, below-base C2 sign (below V1), below-base V2 sign> in the Lanna style or <C1 base, post-base V1 sign, below-base C2 sign (below V1), post-base V2 sign> in the Khuen–Luen style.
  - Does it allow further chaining of another syllable?
  - **A4. Action item for Martin Hosken and Patrick Chew:** Investigate when a consonant sign (including both the regular and medial signs of LA) can be attached to a post-base vowel sign.

- The *Core Specification* is unclear or incomplete about the expected encoding order of many scripts, or specifies something that's different from actual implementations.
- About pre-base vowel signs:
  - for a glyph sequence <pre-base V sign, C1 base, below-base C2 sign>, the encoding order should be <C1 letter, C2 sign, V sign>, no matter the encoding of the C2 sign is <stacker, C2 letter> or an atomic medial C2 sign.
  - If the pre-base V sign is encoded immediately after C1, the theoretical complexity between the base and pre-base V sign can be minimized, but such a <C1 letter, V sign, C2 sign> order diverges from the general <C1 letter, C2 sign, V sign> order for other vowel signs.
- Currently the encoding order affects reordering in certain combinations of a font and a text engine: a pre-base vowel sign must be encoded immediately after the C base, otherwise it won't be reordered.
  - Shaping engines do have inconsistencies.
  - Some fonts are not using the correct script tag. This, depending on the exact shaping engine, may either help with or prevent the pre-base vowel sign reordering.

B.5. Poorly supported character sequence <C1 letter, V sign, C2 sign>

- Not supporting this sequence is a USE deficiency.
  - When either V or C2 is post-base and spacing, the usual workaround <C1 letter, C2 sign, V sign> is not an option.
- There is a confusability between <C1 letter, C2 sign, V sign> and <C1 letter, V sign, C2 sign> when V is not post-base.
  - The cluster pattern needs to block the stacker or consonant sign after a non-post-base vowel sign.
- What about atomically encoding all final consonant signs?
  - Final consonant signs generally are only phonetically distinct from non-final consonant signs.
  - A confusable pair of encodings, <C1 letter, C2 sign> and <C1 letter, final C2 sign>, would be introduce for the structure <C1 base, C2 sign>, where C2 is often not a syllable final consonant.
  - No enough space left in the Tai Tham block or BMP.
- The USE specification can be updated to allow a stacker to follow a vowel sign.
  - It's preferred to treat Tai Tham as special case, so the USE's specification for other scripts is not disturbed.
- Can Tai Tham be supported by another engine instead?
  - Microsoft is not likely to invest in a new engine.

B.6. Tone signs

- Experts' preferences about where the tone sign should be in the cluster pattern:

- CV+CT (the tone sign follows the whole cluster): Andrew's 1st.
- CVT+C (the tone sign immediately follows vowel signs): Andrew's 2nd.
- CTV+C (the tone sign immediately follows the base): Martin.

- When the tone sign is encoded not according to its graphical placement but syllabic role, how can the cluster pattern allow multiple possible placements?
  - For example, if <C1 base, post-base V sign, below-base C2 sign (here below-V), above-base T sign> is attested, there's no way to distinguish two possible placements of T: above C1 vs above V.
  - **A5. Action item for Martin Hosken and Patrick Chew:** Confirm if syllable chaining or other mechanisms can lead to <C1 base, post-base V sign, below-base C2 sign (below V), above-base T sign (above V)>.
  - **A6. Action item for Martin Hosken:** Document the different analyses of CVC and CCV for a syllable ending with -y.
- The USE doesn't currently allow CTV (a tone sign between the base and the vowel sign).

B.7. Khuen–Lue transferable OA BELOW

- When a below-base vowel sign V is applied to a base C1 that already has a below-base consonant sign C2, although the Lanna styles allows a second-level below-base sign, the Khuen–Lue style requires V to be transferred to the right of C2.
  - In such a case, V usually takes an alternative post-base form, which has its own inline advance. However, the vowel sign represented with U+1A6C TAI THAM VOWEL SIGN OA BELOW doesn't have a distinct post-base form.
  - As the vowel sign OA BELOW is simply written to the right of C2 without any inline advance, when the base C1 is followed by another base C3, graphically OA BELOW appears to belong to C3.
  - Note that in common text the spelling context should be able to orthographically clarify which base OA BELOW actually belongs to.
- **A7. Action item for the native users:** Investigate in what order native users write the leading syllable's vowel sign OA BELOW and the following syllable's consonant letter when OA BELOW is transferred to the following syllable's base. (Arloka believes Khuen users write V first.)
- This Khuen–Lue specific behavior can be considered as a stylistic process and be resolved by fonts:
  - The difference in written structures is purely driven by context when the regional variant is known. Thus the two placements can be encoded the same.
  - Richard has an example of this issue, "lɔːt^{3}", on his page ↗ *Tai Tham placement patterns*, showing there are two ways of encoding (CCV.C and CC.CV) because Khuen–Lue fonts are allowed to be smart.
- Or, this behavior may require differentiation in the encoding:

    - Marginal cases always reveal themselves later and can be hard to handle when the encoding model has already been designed to rely on seemingly systematical rules.
    - Users might want to differentiate them within a single regional variant. Note it's typical for such a requirement to be proposed to the UTC.
- What if fonts are required to explicitly differentiate the two encodings, by rendering CCV.C with a spacing V (empty but spacing on the baseline)?
    - Users are presented with explicit difference so they can choose CC.CV instead if no extra spacing is desired.
    - However the text engine can't validate this, and fonts won't be implemented consistently.
- Note that generally in the Myanmar script, if the following base also has a below-base vowel sign, the two clusters need to be kerned so the two vowel signs don't clash.
- *Liang Hai's comment during review:* What if C3 is wide? Is OA BELOW's placement then distinguishable between CCV.C and CC.CV?

B.8. MAI KANG LAI's various placement styles

- Where should the combining mark U+1A58 TAI THAM SIGN MAI KANG LAI be in the cluster pattern for the following placement styles in <C1 base, C2 base>?
    - Lanna: above C2.
    - Khuen–Lue: between C1 and C2.
    - Anomaly: above C1.
- If users are to be allowed to control the placement style:
    - For "between C1 and C2" and "above C1": <C1, MAI KANG LAI, C2>
    - For "above C2": <C1, C2, MAI KANG LAI>
- Otherwise, if fonts are to take full control of the placement style:
    - Always <C1, MAI KANG LAI, C2>
    - **A8. Action item for the Tai Tham font developers:** Investigate whether it is desired to have the multiple styles of placing MAI KANG LAI available as options in a single font, or it is acceptable to have them available only as a font-to-font variation.
- Note that while Tai Tham representative glyphs are generally in the Khuen–Lue style, the representative glyph of MAI KANG LAI takes the Lanna style because this character is only distinct from another sign in Lanna.
    - Annotate this in the names list.
    - **A9. Action item for Martin Hosken and Patrick Chew:** Find out from the original proposal what the contrast is between MAI KANG LAI and the other sign in Khuen–Lue. Can they be unified in Khuen–Lue?
- A potential change: MAI KANG LAI can have InSC = Consonant_Prefixed so the USE can reorder it by default.
    - When the repha reordering is not desired, it can be disabled in fonts.
    - The representative glyph should then have a dashed box.

- The *Core Specification* should then document this special behavior.
- Note that a font can't opt-in the repha reordering for a normal combining mark with a "sub mark by mark;" rule in the feature rphf, because a USE cluster can't start with a combining mark.
- The USE specification currently says "Note that the category REPHA is not currently supported by USE."
- **A10. Action item for Andrew Glass:** Investigate the progress of implementing REPHA in Microsoft's own USE implementation.
- **A11. Action item for Andrew Glass:** Provide exemplar fonts that demonstrate how to implement the multiple styles of placing MAI KANG LAI, and how to stop a shaping engine from reordering MAI KANG LAI when the reordering is not desired.

⋆ EOF ⋆