L2/22-068

TO: UTC

FROM: Deborah Anderson, Ken Whistler, Roozbeh Pournader, and Peter Constable[1]

SUBJECT: Recommendations to UTC #171 April 2022 on Script Proposals

DATE: April 15, 2022

The Script Ad Hoc group met on February 11 and 17, March 11 and 19, and April 8, 2022, in order to review proposals. The following represents feedback on proposals that were available when the group met.

Table of Contents

[1] Also participating were Fred Brennan, Craig Cornelius, Simon Cozens, Lorna Evans, Andrew Glass, Manish Goregaokar, Liang Hai, Ned Holbrook, John Hudson, Richard Ishida, Marek Jeziorek, Frank van de Kasteelen, Jan Kučera, Norbert Lindenberg, Kamal Mansour, Karan Misra, Lisa Moore, Michel Suignard, Lawrence Wolf-Sonkin, and Ben Yang. The text for the comments and recommendations was based on notes taken by Debbie Anderson, and Jan Kučera and comments from Norbert Lindenberg.

I. EUROPE

1 Latin

1a Capital Rams Horn

Action: For UTC discussion and decision

Document: L2/22-045 Proposal to add capital rams horn – Jacquerye

Comments: We reviewed this revised document for one Latin character. An earlier version of the proposal (L2/21-205) was seen by the SAH, with comments contained in L2/21-174.

In response to the SAH comments, the author has moved the character to a different code point (which the SAH recommended) and included properties.

We recommend approval of LATIN CAPITAL LETTER RAMS HORN.

Recommendations: We recommend the UTC make the following disposition:
SAH-UTC171-R1: Accepts U+A7CB LATIN CAPITAL LETTER RAMS HORN for a future version of the standard. (Reference: L2/22-045)

The following actions are recommended:
Action Item for Ken Whistler: Update the Pipeline to include U+A7CB LATIN CAPITAL LETTER RAMS HORN (Reference: L2/22-045)
Action Item for Debbie Anderson and Denis Moyogo Jacquerye: Forward a font to Michel Suignard for U+A7CB LATIN CAPITAL LETTER RAMS HORN. (Reference: L2/22-045)

1b Latin Small Letter Turned O Open-o etc.

Action: FYI with action to record

Document: L2/22-097 On LATIN SMALL LETTER TURNED O OPEN-O and LATIN SMALL LETTER TURNED O OPEN-O WITH STROKE -- Denis Moyogo Jacquerye

Comments: We reviewed this proposal, which proposed options for handling apparent errors in U+AB43 LATIN SMALL LETTER TURNED O OPEN-O and U+AB44 LATIN SMALL LETTER TURNED O OPEN-O WITH STROKE. These two characters were approved by the UTC, based on the revised proposal for "Teuthonista" phonetic characters (L2/11-202), and were published in Unicode version 7.0.

The characters were originally proposed on the basis of one example from Bremer 1898, but the character names and glyphs do not represent the description found in Bremer 1893, i.e., the initial element is an *a*, not a *turned open o*. It is not clear from the examples whether the *a* is a *two-story a* or a Latin *alpha*.

Two options for changing the glyphs are proposed in this document:
a. Change the glyphs to be a ligature of *two-story a* with an *o* / *o* with stroke
b. Change the glyphs to be a ligature of a Latin *alpha* with an *o* / *o* with stroke.

The following points were raised:

- Number the pages.
- Of the two options mentioned, in our opinion only the second is feasible, since a glyph change for the first would mean the glyph would be identical to U+A735 LATIN SMALL LETTER AO. We recommend the author revise the document, briefly mention option 1, but propose option 2.
- Include a recommended annotation for U+AB43 and U+AB44 (i.e., "this is actually a ligature of …," referring to the first element as the phonetic symbol U+0251 LATIN SMALL LETTER ALPHA).
- The recommended update to UTN #27 is reasonable, but refer to the first element as the phonetic symbol U+0251 LATIN SMALL LETTER ALPHA.

Recommendations: We recommend the UTC make the following disposition:
Notes this document; the SAH comments have already been conveyed to the author.

The following actions are recommended:
Action Item for Debbie Anderson: Confirm the glyph change is acceptable to those involved on the revised Teuthonista proposal L2/11-202. (Reference: section 1b of L2/22-068)

1c Latin Small Letter Closed Open E etc.

Action: For UTC discussion and decision

Document: L2/22-071 Proposal to revise the glyphs of LATIN SMALL LETTER CLOSED OPEN E and LATIN SMALL LETTER CLOSED REVERSED OPEN E -- Denis Moyogo Jacquerye

Comments: We reviewed this request to change the representative glyphs for two IPA symbols, U+025E LATIN SMALL LETTER CLOSED REVERSED OPEN E and U+029A LATIN SMALL LETTER CLOSED OPEN, so they are changed from a Greek-like appearance to one that is more Latin-like, i.e., so the vertical stems are thicker than the horizontal stems. This Latin-like appearance is already reflected in the glyph for U+1078F MODIFIER LETTER SMALL CLOSED REVERSED OPEN E. According to the proposal, IPA symbols borrowed from Greek are typically re-designed to reflect the design of Latin letters.

John Hudson voiced his approval of this change, noting that Brill requested this approach to the design of their glyphs for fonts used in academic publishing. The SAH also was in agreement with the requested change.

Recommendations: We recommend the UTC make the following disposition:
SAH-UTC171-R2: Accepts the glyph changes for U+025E LATIN SMALL LETTER CLOSED REVERSED OPEN E and U+029A LATIN SMALL LETTER CLOSED OPEN E for correction in Unicode 15.0. (Reference: L2/22-071)

The following action is recommended:
Action Item for Debbie Anderson: Ask Michael Everson for a font for U+025E LATIN SMALL LETTER CLOSED REVERSED OPEN E and U+029A LATIN SMALL LETTER CLOSED OPEN E with changes as described in L2/22-071.

2 Todhri

Action: For UTC discussion and decision

Documents:
L2/20-188r2 Proposal for encoding the Todhri script in the SMP of the UCA – Everson (SEI)
L2/22-074 Todhri Encoding Options -- Pournader

Comments: The SAH had recommended approval of the Todhri script based on L2/20-188r2 in its Oct. 2020 recommendations, but the UTC assigned an Action Item (165-A24) to Roozbeh Pournader and Debbie Anderson to discuss the decomposition of U+105C9 TODHRI LETTER EI and U+105E4 TODHRI LETTER U, before the script would be considered ready for approval.

Document L2/22-074 presents the three options (below), and describes pros and cons of each.

1. Encode the letters EI and U as characters, with no canonical decomposition.

2. Encode the letters EI and U as characters, with canonical decompositions to <I, COMBINING DOT ABOVE> and <O, COMBINING DOT ABOVE>.

3. Do not encode EI and U as characters, and represent them as the sequences <I, COMBINING DOT ABOVE> and <O, COMBINING DOT ABOVE>

During the past year, the Script Ad Hoc has *not* been recommending “Do Not Use” tables for decompositions, since they aren’t being implemented consistently. As a result, Option 1 is not a viable choice. Option 2, with atomic characters for EI and U with canonical decompositions would be best for search and collation. Option 3 is also a possible option, but it would not be as good for search and collation, and tailoring or contractions would be required for DUCET.

In our opinion, option 2 is best, that is, atomically encode U+105C9 TODHRI LETTER EI and U+105E4 TODHRI LETTER U, but provide canonical decompositions.

We recommend having the atomic characters on a keyboard, without a separate key for COMBINING DOT ABOVE.

If the UTC agrees with this decision, we recommend the UTC accept the 52 Todhri characters.

Recommendations: We recommend the UTC make the following disposition:
SAH-UTC171-R3: Accepts 52 Todhri characters in a new Todhri block (U+105C0..U+105FF) for encoding in a future version of the standard, but amending the properties on page 6 of L2/20-188r2 with the following two decompositions and changing the general category property for all the characters from “Ll” to “Lo”:

105C9;TODHRI LETTER EI;Lo;0;L;105D2 0307;;;;N;;;;;
105E4;TODHRI LETTER U;Lo;0;L; 105DA 0307;;;;N;;;;;
(Reference: section 2 of L2/22-068 and L2/20-188r2)

The following actions are recommended:
Action Item for Ken Whistler: Update the Pipeline to include 52 Todhri characters. (Reference: section 2 of L2/22-068 and L2/20-188r2)
Action Item for Debbie Anderson and Michael Everson: Send Michel Suignard a font. (Reference: section 2 of L2/22-068 and L2/20-188r2)

II. AFRICA

3 Garay

Action: For UTC discussion and decision

Documents:
L2/22-048 Consideration of the encoding of Garay with updated user feedback (revised) -- Rovenchak
Related documents:
L2/16-069 Proposal for encoding the Garay script in the SMP of the UCS -- Everson

Comments: We reviewed this document on Garay, a script used to write Wolof. This document is a revision of L2/22-030 (dated January 18 2022), with five additional characters added to the repertoire: two case pairs of obsolete characters ("OLD KA" and "OLD NA") and Sukun, used to indicate a consonant with no following vowel.

This proposal is the culmination of several years of back-and-forth with the user community. A meeting took place in 2017 between Chuck Riley and the script creator (L2/17-322) in order to get answers to questions that had been earlier posed by the SAH. Calls with the creator (who unfortunately died in 2021), his son, and others have taken place since 2019 in order to carefully review the proposal and get additional information.

The following comments were raised during discussion on L2/22-048:

- Written material in the script includes a primer and a translation of the Quran. Additional evidence was provided in the earlier proposal by Michael Everson, L2/16-069.
- Interest in encoding the script has been expressed by the head of Boston University's African Collection and the director of the West African Research Association at Indiana University (mentioned in report from 2020, L2/20-067).
- The latest document provides enough information to confirm the bidi_class property.
- The script is not "newly invented," but dates from 1961 and has reportedly been taught for over 50 years. In our view, it meets the bar set for a script with historical value.

Recommendations: We recommend the UTC make the following disposition:
SAH-UTC171-R4: Accepts 69 Garay characters in a new Garay block (U+10D40..U+10D8F) for encoding in a future version of the standard. (Reference: L2/22-048)

The following actions are recommended:
Action Item for Ken Whistler: Update the Pipeline to include 69 Garay characters. (Reference: L2/22-048)
Action Item for Debbie Anderson and Andrij Rovenchak: Send Michel Suignard a font. (Reference: L2/22-048)

III. MIDDLE EAST

4 Arabic

4a Behdini Kurdish Heh

Action: FYI with action to record

Document: L2/21-087 Regarding the Behdini Kurdish Heh -- Evans

Comments: We reviewed this April 2021 proposal. Review of the document has been on hold while the Script Ad Hoc awaited feedback from an expert; input from the expert was recently received.

The following SAH comments from L2/21-073 summarize the three possibilities for representing Behdini Kurdish *heh*:
- Consider it a glyph variant of U+0647 ARABIC LETTER HEH
- Consider it a glyph variant of U+06BE ARABIC LETTER HEH DOACHASHMEE
- Encode a new character.

Although the community has been using U+0647, we agreed that U+06BE ARABIC LETTER HEH DOACHASHMEE should be employed, since it provides the needed shapes, i.e., two "eyes" in all forms. Additional information is needed before making a recommendation for Sindhi.

Recommendations: The following action is recommended:
Action Item for Lorna Evans: Adjust the following text in section 9.1 of TUS, under "Letter heh" (p. 388, Unicode version 14.0): "U+06BE ARABIC LETTER HEH DOACHASHMEE is used to represent any *heh*-like letter that appears with stems at both sides in all contextual forms." Also, document that Behdini Kurdish *heh* should be represented with U+06BE. (Reference: section 4a of L2/22-068)

4b Combining Alef Overlay

Action: For UTC discussion and decision

Document: L2/22-047 Arabic: Proposal to encode Arabic Combining Alef Overlay used in Quran published in Libya -- Lateef Sagar

Comments: We reviewed this proposal to add one Arabic character, ARABIC COMBINING ALEF OVERLAY, which occurs in some Qurans, especially in Libya. The character was discussed earlier by the Script Ad Hoc (see Jan 2022 SAH Recommendations L2/22-023 and the October 2021 SAH Recommendations L2/21-174). It also appeared in earlier documents (X7 in L2/19-306 and characters #28 and #29 in L2/15-329).

The revised proposal includes justification for a combining character (as opposed to a precomposed, atomic character) and a name that is acceptable.

After some discussion about the CCC value, it was decided it should be 0 (instead of 1, which is the CCC for a typical Latin overlay, or 35, the CCC of superscript alef, which has the same semantics as the proposed character). The rationale for choosing CCC=0 is because in cases of a typical sequence such as <*lam*, a *tashkeel*, COMBINING ALEF OVERLAY, another *tashkeel*>, anything other than CCC=0 could result in the loss of information when the string is normalized, with the relationship of the *tashkeel* to their base lost.

Once this character is approved (with CCC=0) for a specific Unicode version, UTR#53 will then need to be updated, as well as text in the Core Spec.

Recommendations: We recommend the UTC make the following disposition:
SAH-UTC171-R5: Accepts U+10EFC ARABIC COMBINING ALEF OVERLAY for a future version of the standard. (Reference: L2/22-047)

The following actions are recommended:
Action Item for Ken Whistler: Update the Pipeline to include U+10EFC ARABIC COMBINING ALEF OVERLAY. (Reference: L2/22-047)
Action Item for Lorna Evans: Forward a font to Michel Suignard for U+10EFC ARABIC COMBINING ALEF OVERLAY. (Reference: L2/22-047)

4c High Hamza

Action: FYI with action to record

Document: L2/22-051 Proposal to Encode ARABIC LETTER THREE QUARTER HIGH HAMZA for Jawi -- Ahmad Ali A. Karim, et al.

Comments: We reviewed this request for one Arabic character, THREE QUARTER HIGH HAMZA, which the authors state is used to write Jawi text (Malay language in the Arabic script). The authors mention the proposed character could be considered a variant of U+0621 ARABIC LETTER HAMZA (except the glyph is raised in THREE QUARTER HIGH HAMZA) or U+0674 ARABIC LETTER HIGH HAMZA (except the glyph should be larger and positioned lower). U+0674 currently has an annotation “Kazakh, Jawi” which was added based on L2/20-289. The authors report users have commonly been superscripting U+0621.

The topic of a THREE QUARTER HIGH HAMZA character was raised earlier at a Unicode conference and an October 2020 email thread on the Unicode list started by Yaya (“Question for Malay Jawi letter in

Unicode – three quarter hamza”). The email thread included an attachment by Yaya. The attachment mentions that some fonts, including notably the Amiri font, displays U+0674 ARABIC LETTER HIGH HAMZA at three quarters height for Jawi.

One member of the SAH thought the character should be disunified from U+0674, noting that HIGH HAMZA was originally intended to be a second component in Kazakh digraphs, not a stand-alone letter.

However, the general consensus amongst the group was to unify THREE QUARTER HIGH HAMZA with U+0674 HIGH HAMZA, and to highly encourage font designers to support the glyph with the expected shape and positioning for Jawi, employing a language tag or creating a Jawi-specific font. Encoding a new character would take several years, and text representation would be inconsistent, causing other problems for users.

Recommendations: The following action is recommended:
Action Item for Rick McGowan: Relay the comments in Section 4c of L2/22-068 to Ahmad Ali A. Karim, et al., authors of L2/22-051.

4d Sindhi HEH

Action: FYI with action to record

Documents:
L2/22-052 Regarding the Sindhi Heh – Evans
L2/22-053 Sindhi Heh -- Mansour

Comments: Sindhi and several other languages have a normal /h/ consonant as well as aspirated consonants. The glyph shapes for the two *heh's* do not exactly match those of U+0647 ARABIC HEH or U+06BE HEH DOACHASHMEE, which has led to confusion by users about the appropriate character. In addition, Sindhi represents words with Arabic language *heh* with a different glyph from the Sindhi *heh.* Lastly, there are instances of a final schwa, aspirated schwa, or no sound in word-final position that Evans reports is typically represented by U+06C1 HEH GOAL.

An examination of the examples of Sindhi HEH from Mansour, Evans, and a 2020 primer are not consistent in handling the *heh's*:
(1) In Evans' document, Sindhi *heh* [h] has a knotted form, but Mansour's example from Mr. Majid Bhurgri – a well-respected user of the script – shows a butterfly form.
(2) In Evans' document, the aspirated *heh* [^{h}] has the butterfly form, but Mr. Bhurgri's example has the knotted form;
(3) In the 2020 primer, both *heh*'s ([h], [^{h}]) are represented with the butterfly form.

Because of the inconsistency, it is difficult to know what to do. Unicode already offers three different *heh*'s. Do they provide enough nuance required by users? How are children being instructed in schools in primers?

To try make progress and get more information, Lorna Evans will see if she has contacts with any of other languages that have *heh*'s and aspirated consonants and see how those languages handle them.

As an update, Lorna reports that she has located 5 primers for other languages that also have a normal /h/ consonant and aspirated consonants (like Sindhi).

In addition, Kamal Mansour offered to send Mr. Bhurgri phonetic transliteration of words with *heh*'s and ask him to write out words. (Kamal Mansour recently relayed that he has followed up on this action.)

Recommendations: The following action is recommended:
Action Item for Kamal Mansour and Lorna Evans: Try to contact the Sindhi Language Authority for their input, if possible. (Reference: section 4d of L2/22-068)

IV. SOUTH AND CENTRAL ASIA

5 Gurung Khema

Action: FYI with action to record

Document: L2/22-096 Proposal to Encode the Gurung Khema Script in Universal Character Set – Biswajit Mandal

Comments: We reviewed this revised document. The user community is now involved in the review of the proposal, with experts from Nepal and Sikkim. The script has undergone development; the last revision to the script took place in 2019 (Section 3.5).

The following summarizes the comments (by section, in bold):

Background.
The Sikkim State Government reportedly recognized Gurung Khema as an official language in 1995, but in figure 4 the text has the date as 1997. Correct or explain the discrepancy in the dates. Also, if possible, provide a copy of the official Sikkim State Government declaration or a news article confirming that the script was official in Sikkim. (The text refers to figure 4 as evidence of the script's official status, but figure 4 only mentions the official date of the language.)

Remove the comment about the Roadmap request; Khe Prih has been removed in the March 2022 posted version. There is no requirement a script in Unicode be official or that there be only one script per language. In general, proposal authors should start by using XXXX0, XXXX1, etc. in the proposal, and request code points. If the script is deemed eligible for encoding, the UTC will recommend code points, the proposal author will be notified, and the script's allocation will appear on the Roadmap.

Section 3.2.2.
Lãĩl̥homā (anusvara) should not be unified with U+030C COMBINING CARON; it should be separately encoded so its Indic syllabic and positional categories can be defined (i.e., Bindu / Top). Also, specify its position in collation (Section 7). The description of the positioning of the glyph should not be based on problems in a font. Instead, focus on the proper glyph position. An appendix to a proposal can consider how it should look versus how it may look in some fonts. An example from print material may be helpful and/or a description.

State whether the anusvara appears on vowels and consonants, and point to a figure which shows this behavior.

Section 3.2.3.
The set of examples on the bottom of page 4 (*khya*, *khva*, *khra*) should include the sequence of characters (that is, the explicit encoding order), similar to the sequences shown for CONSONANT SIGN MEDIAL HA/VA/YA on the same page. In the sequences, MEDIAL HA should be last. Also, mention that when MEDIAL HA is used with another medial, MEDIAL HA always comes second.

CONSONANT SIGN MEDIAL RA: This character should be separately encoded, so its Indic syllabic and positional categories can be defined (Consonant_Medial / Bottom). Also, specify its position in collation (Section 7).

Section 3.4.
Add a comment that additional characters can be proposed to represent the older texts at a later time, if needed.

Section 6.
Fix the numbering, as Section 6 on page 7 starts with “5.1” and continues through page 9, when a new Section 6 starts.

Change the decomposition order for SIGN EE to be: LENGTH MARK + SIGN I. (The order goes outwards from the base.)

Add a decomposition to the list on page 10 for SIGN U showing that it decomposes to SIGN AA + SIGN AA.

SIGN OO has three parts, but decompositions should only have two, so change the decomposition for SIGN OO to be: SIGN UU + SIGN I (as opposed to SIGN AA + SIGN EE). Note that the first part should be the composite part and the second part be simpler. (The decomposition in the Jan. 29 2022 version of the proposal was correct for SIGN OO, but in that version, SIGN II should be removed, as it should have no decomposition. The Jan 29 version was also missing the decomposition for SIGN EE.)

Add the above decomposition changes to the properties section on page 11 and names list on page 14.

Properties
The Indic syllabic category for LETTER A should be Vowel_Independent, as it is used as such and not just as a vowel carrier.

There should be a description of line breaking requirements: Does the script use spaces to separate words, does it break at word boundaries or are breaks within a word allowed, are hyphens used? (This information can be moved into a separate paragraph in the main body of the proposal, cf. Lampung proposal L2/22-044 where it appears in a separate section.)

The Line_Break category of the medial consonants should be CM (not SA), because the base letters use AL. The category SA indicates the South East Asian style of context analysis, while AL indicates the Western style. These two styles can’t be combined in one script.

Chart and names list
Move the VOWEL LENGTH MARK up in the chart to U+16129 so it is grouped with Vowel Signs. In the names list, group it with Dependent Vowel Signs, and move it from section 5.4 to 5.3 in the descriptive text of the proposal.

Change the annotation for GURUNG KHEMA SIGN THOLHOMA from “virama” to “vowel killer, always rendered visibly” (which is a more precise definition).

Figures
Re-insert the image from figure 11 from earlier version of proposal L2/21-145. This answers the question posed by the SAH whether a book has been written in the script (rather than one about the script).

Figure 14 has several lines ending with what looks like C, i.e. number 2 in the script. Is this actually meant to be number 2?

In 5 Usage / Educational Materials, the last *[see fig.16]* should be fig 17.

ISO Proposal Form
List those involved in the review of the proposal and whether they live in Nepal or Sikkim.

Recommendations: The following action is recommended:
Action Item for Lorna Evans or Debbie Anderson: Forward the comments from section 5 of L2/22-068 to Biswajit Mandal, the author of L2/22-096.

6 Kirat Rai

Action: For UTC discussion and decision

Document: L2/22-043 Proposal to Encode Kirat Rai script in the Universal Character Set-- Mandal and Evans

Comments: We reviewed this proposal, which builds on the preliminary proposal L2/21-132. The proposal has gone through several intermediate revisions seen by the Script Ad Hoc.

After extensive discussion, the model we recommended (and adopted in this proposal) is to encode three vowel signs AI, O and AU atomically and give them canonical decompositions. In this approach, no Do Not Use table is necessary.

The posted revised version has incorporated various feedback from the Script Ad Hoc session and we consider it ready to be recommended for approval.

Recommendations: We recommend the UTC make the following:
SAH-UTC171-R6: Accepts 58 Kirat Rai characters in a new Kirat Rai block that extends from U+16D40..U+16D7F for a future version of the standard. (Reference: L2/22-043)

The following actions are recommended:

Action Item for Ken Whistler: Update the Pipeline to include 58 Kirat Rai characters. (Reference: L2/22-043)
Action Item for Lorna Evans: Forward a font to Michel Suignard for 58 Kirat Rai characters. (Reference: L2/22-043)

7 Tulu-Tigalari

Action: FYI with action to record

Document: L2/22-075 Letter on Tulu-Tigalari from Dr. Guru Prasad, Director, KaNaada Phonetics Pvt. Ltd. -- Guru Prasad

Comments: We reviewed this document, which expressed concerns about the TULU-TIGALARI CONJOINER. The author advocates relying on virama, in part because it will allow easy transition between Tulu-Tigalari and other ISCII-based Indic scripts for editors, keyboards, and fonts. For half-consonants, the author recommends use of ZWJ/ZWNJ or ZWSP.

The question as to why a conjoiner is proposed – and not a virama — is reasonable, since users of the main ISCII scripts are used to an encoding model with a virama. However, the Script Ad Hoc has moved away from the ISCII encoding model, separating the functions of "virama" into two characters: a stacker for conjuncts and a vowel-killer.

The ISCII model of virama as a conjoiner has created several issues in scripts like Devanagari, and Malayalam (the direct analog to Tulu-Tigalari). Particularly, in Malayalam, this has led to several ad-hoc solutions being proposed to fix them and has unnecessarily increased the complexity of encoding Malayalam texts. Even fairly common script behavior requires the extensive use of invisible ZWJ/ZWNJ characters and idiosyncratic Unicode sequences. Given that Tulu-Tigalari is a fresh encoding without any legacy baggage, separating the conjoiner and the virama (along with the dot reph) simplifies the encoding model and provides the user with clean input sequences that can faithfully render various visual forms occurring in manuscripts without the use of invisible characters using straightforward and intuitive sequences. (Refer to sections 5.4, 5.5 and 5.6 in L2/22-031.)

Khmer and Myanmar already use such models, and transliterators have been implemented that could perform round-trip between such scripts and ISCII-based Unicode scripts seamlessly. They just require a few regex replacements that any programmer should be able to implement without much effort. For users who are used to having a single virama key on keyboards, and expect input of a virama to either automatically combine with an adjacent consonant into a conjunct form or half form, or to remain a visible virama mark, a smart keyboard can automatically choose between conjoiner and visible virama based on the surrounding characters. For users who need precise control, such as manuscript experts who need to exactly replicate the appearance of manuscript text, two separate keys for conjoiner and visible virama may be provided. On-screen keyboards can instead offer direct input of conjunct forms, where a single keystroke creates the sequence conjoiner-consonant.

Recommendations: The following action is recommended:
Action Item for Debbie Anderson: Forward comments from section 7 of L2/22-068 to Guru Prasad, the

author of L2/22-075.

V. SOUTHEAST ASIA, INDONESIA, AND OCEANIA

8 Balinese

Action: For UTC discussion and decision

Document: L2/22-059 Proposal to Encode Three Balinese Punctuation Marks -- Rikza F. Sh., Aditya Bayu Perdana

Comments: We reviewed this request for three Balinese punctuation marks found in older documents. The document provides good evidence for the two proposed characters INVERTED CARIK SIKI and INVERTED CARIK PAREREN, showing contrast between them and their non-inverted versions, already encoded at U+1B5E and U+1B5F. Examples are also provided showing BALINESE PANTI BAWAK in contrast to its untruncated version, PANTI, already encoded at U+1B5A.

The code points are acceptable. If these three characters are approved, one open slot will remain in the Balinese block.

Recommendations: We recommend the UTC make the following disposition:
SAH-UTC171-R7: Accepts 3 Balinese punctuation characters
1B4E BALINESE INVERTED CARIK SIKI
1B4F BALINESE INVERTED CARIK PAREREN
1B7F BALINESE PANTI BAWAK
for encoding in a future version of the standard. (Reference: L2/22-059)

The following actions are recommended:
Action Item for Ken Whistler: Update the Pipeline to include 3 Balinese punctuation characters. (Reference: L2/22-059)
Action Item for Debbie Anderson to send Michel Suignard a font for 3 Balinese punctuation characters. (Reference: L2/22-059)

9 Javanese

Action: For UTC discussion with action to record

Document: L2/22-041 Javanese orthographic syllables – Lindenberg

Comments: We discussed this document which describes the order of syllables for Javanese. Such information is useful for developers of fonts, rendering systems, spell checkers, smart keyboards and other text-producing software that can identify character sequences that are not in standard encoding order. This document also addresses an error report from David Corbett on “Bad Javanese BNF” from 2019-12-08, regarding the BNF for Javanese in the Core Spec.

While such descriptions are useful, concerns were voiced during discussion about scoping such a description for other scripts in the Core Spec, and the amount of work it would entail for the Editorial Committee.

To make such orthographic syllable descriptions consistent – and lessen the load on Editorial Committee – we recommend interested members work together to extract all pieces from the Core Spec that describe the order of syllables in order to address inconsistencies. The final location of such descriptions would likely not be the Core Spec, but could be a new UTR or UAX.

To address the error noted by David Corbett, we recommend the entire section "Ordering of Syllabic Components" be removed.

Recommendations: The following action is recommended:
Action Item for Julie Allen and the Editorial Committee: Remove the entire section "Ordering of Syllabic Components" from the Javanese block introduction. (Reference: section 9 of L2/22-068 and L2/22-041)

10 Lampung

Action: FYI

Documents:
L2/22-044 Revised proposal to encode the Lampung script – Pandey
L2/22-057 Comments on Revised proposal to encode the Lampung script – Aditya Bayu Perdana, et al.
L2/22-058 Comments on Revised proposal to encode the Lampung script – Febri Muhammad Nasrullah

Comments: We discussed feedback from Febri Muhammad Nasrullah and Aditya Bayu Perdana (et al).

The following comments were raised during discussion:

- Character names: We recommend using the character names for consonants and vowels contained in the February 18, 2022 version of the proposal, except for VOWEL SIGN EA, for which we recommend VOWEL SIGN EU.
- Digits: We agree that removing digits is acceptable, given the information currently available.
- Punctuation:
  o The following two characters are acceptable, but we recommend the following character names be changed from "Sign" to "Mark", so they are: START OF TEXT MARK, END OF TEXT MARK.
  o Remove the following four punctuation characters from the list of characters for encoding for now (but note the recommended change in two names from "Sign" to "Mark"): FULL STOP, FULL COMMA, EXCLAMATION MARK, QUESTION MARK.
- On RA/HA/GHA: Annotate GHA, noting that it is not in current use.
- Chart/Names list:
  o Remove the subhead "Virama" and replace it with "Sign". (The term "virama" is now deemed too vague, since it can be used for different functions.)
  o Change the annotations for LAMPUNG SIGN NENGEN to read "vowel killer, does not produce

conjuncts, always rendered visibly.”

- o Question: Does LAMPUNG SIGN NENGEN appear traditionally after dependent vowel signs or in another location in the sort order? Its current position appears fine, but there was a question whether it had a traditional location in sorting.

Note: The above comments have been forwarded to Anshuman Pandey, the Lampung proposal author.

Recommendations: The UTC notes this document, but take no further action.

11 Myanmar

Action: For UTC discussion and decision

Document: L2/22-046 Proposal to encode numerals for Eastern Pwo Karen and Pa’O – Mitchell

Comments: We reviewed this proposal to add 20 numerals used to write Eastern Pwo Karen and Pa’O. Evidence is contained in the proposal, and the author has confirmed they are in current use. The author mentions (page 1) that additional numerals may be needed for Khamti but further investigation is needed. As a result, they are not yet proposed (but examples are contained on pages 15-16).

Because additional Myanmar characters are expected and only one spot is available in the Myanmar Extended-B block, we recommend the characters be located as a set in a new Myanmar Extended-C block that extends from U+116D0..U+116FF.

Recommendations: We recommend the UTC make the following disposition:
SAH-UTC171-R8: Accepts 20 Myanmar numerals (U+116D0…U+116E3) for Eastern Pwo Karen and Pa’O in a new Myanmar Extended-C block, with range U+116D0..U+116FF for a future version of the standard. (Reference: L2/22-046)

The following actions are recommended:
Action Item for Ken Whistler: Update the Pipeline to include 20 Myanmar numerals. (Reference: section 11 of L2/22-068 and L2/22-046)
Action Item for Debbie Anderson and Ben Mitchell: Forward a font for 20 Myanmar numerals to Michel Suignard. (Reference: L2/22-046)
Action Item for Ken Whistler and Debbie Anderson: Confirm the new Myanmar Extended-C block (U+116D0..U+116FF) is incorporated into the Roadmap. (Reference: section 11 of L2/22-068 and L2/22-046)

12 Tai Don

Action: FYI with action to record

Documents:
L2/22-098 Updated proposal to encode the Tai Don script in UCS -- Kushim Jiang
L2/22-099 Comments on Updated proposal to encode the Tai Don script in UCS – Jim Brase

Related documents:
L2/20-207 Preliminary proposal to encode the Tai Khao script in UCS -- Kushim Jiang
L2/20-208 A response to Kushim Jiang, “Preliminary proposal to encode the Tai Khao script in UCS" -- Jim Brase
L2/08-217 Writing Tai Don: Additional characters needed for the Tai Viet script – Jim Brase
L2/06-041 Unified Tai Script for Unicode. -- Ngô Trung Việt, Jim Brase.

Comments: We reviewed the latest proposal by Kushim Jiang for Tai Don, L2/22-098, and comments on the proposal by Jim Brase (L2/22-099), author of the Tai Viet proposal. In 2021, the Script Ad Hoc had reviewed an earlier version of the proposal, L2/20-207, from the same author (Kushim Jiang), but in that document the script was called “Tai Khao.”

- Make sure that for each proposed character, there is at least one example of usage provided in a line of text, as opposed to just a character chart. While character charts are useful, evidence in running text provides us with a better understanding of contrastive usage, and will also help with future font development. For example, no evidence was provided for the punctuation marks in Chart 6.
    - Why are XX5E and XX5F disunified?
- The encoding model should be clearly identified. If it varies from Tai Viet (which has visual order), justification is needed.
- Relying solely on phonetic differences to justify character repertoire of the different “styles” (Brase’s terms for Jinping vs. Lai Chau) is not considered a solid rationale.
- The general category properties for combining marks should be corrected from Lo to the appropriate general category.
- It was noted that two rules on the bottom of page 2 are inconsistent with the Unicode Standard: combining marks should follow base characters (see p. 55 of Chapter 2 of *TUS* “Sequence of Base Characters and Diacritics”).
- In our opinion, additional research into Tai Don is required. Comparison of the details in Table 3.2 of L2/08-217 and Chart 1 reflect some discrepancies: i.e., the glyphs for LOW BO in L2/22-207 appear in Table 3.2 of L2/08-217 as HIGH BO. Careful comparison will help clarify details about Tai Don and provide insight whether unification of Tai Don with Tai Viet should be recommended.

Recommendations: The following actions are recommended:
Action Item for Debbie Anderson: Forward the comments in section 12 of L2/22-068 to Kushim Jiang, the author of L2/22-098.
Action Item for Lawrence Wolf-Sonkin and Liang Hai: Work with Kushim Jiang, the author of L2/22-098, to improve the proposal for Tai Don. (Reference: Section 12 of L2/22-068)

13 Western Cham

Action: FYI with action to record

Documents:
L2/22-049 Western Cham Resolution Meeting -- Hosken
L2/22-050 Response to Western Cham Resolution Meeting Report (L2/22-049) – Jorge López Cortina

Not yet reviewed by Script Ad Hoc:
L2/22-095 Proposal to Encode Western Cham 2021 – Alberto Pérez Pereiro et al.

Related document:
L2/20-061R3 Final Proposal to encode Western Cham (rev. 3) - Hosken

Comments: We reviewed L2/22-049, a report from Martin Hosken describing a meeting from June 2021 with Western Cham leaders, and L2/22-050. a response from Jorge López Cortina to Hosken's document. The meeting report suggests all attendees approved of the Western Cham proposal by Hosken (L2/20-061R2), but the response reports that Leb Ke, an acknowledged expert on the script, opposed the proposal (referring to the latest proposal, L2/20-061R3).

Background: The Western Cham proposal has been stalled, in part because the two communities do not agree on the repertoire. No single unified proposal has yet been submitted that includes all characters requested by the various groups using the script.

After discussion on how to proceed, we recommended Norbert Lindenberg write to Patrick Chew, an intermediary who understands the issues, proposing Chew (or CLAC) write a proposal requesting the additional characters beyond those in Martin Hosken's proposal, or provide a critique of Martin Hosken's proposal, identifying any issues not related to missing characters or concerns about the model. A third alternative is for CLAC to write a separate script proposal, and Chew (or others) to integrate the Hosken and CLAC repertoire into one. In the meantime, a new proposal has been submitted by Alberto Pérez Pereiro et al. (L2/22-095). Patrick Chew is doing an analysis between the new proposal and the Martin Hosken's proposal (L2/20-061R3). (Norbert Lindenberg has done the action mentioned above.)

Recommendations: The following action is recommended:
Action Item for Patrick Chew: Prepare a document that analyzes the differences between L2/22-095 and L2/20-061R3. (Reference: section 13 of L2/22-068)

VI. SYMBOLS

14 POLISH ZŁOTY SIGN

Action: FYI with action to record

Document: L2/22-092 Proposal to encode POLISH ZŁOTY SIGN -- National Bank of Poland

Comments: We discussed this proposal for a Polish złoty sign. The proposal mentions a special code for złoty symbol on the Mazovia computer from the 1980s and shows a Polish zloty symbol on two typewriters from the 1990s.

In our opinion, the Polish złoty sign can already be represented using a sequence of existing characters: <007A, 0142> for lowercase and for uppercase <005A, 0141>. Using this sequence will ensure character stability and support existing data.

Recommendations: We recommend the UTC make the following disposition:
Action Item for Rick McGowan: Forward the comments in section 14 of L2/22-068 to the author of L2/22-092.

VII. PUBLIC REVIEW FEEDBACK AND ALPHA 15.0 FEEDBACK

15 Arabic LAM-LAM-HEH

Action: FYI with action to record

Document: L2/22-063 Comments on Public Review Issues (Subject: Kurdish language problems with the Arabic Script)

Comments: We reviewed the comments submitted by Halbast Abdullah on Wed Jan 26 10:38:02 CST 2022 regarding the automatic ligation of *lam-lam-heh* in fonts, which is a known problem for Kurdish (as well as many other languages).

Background: Some fonts ligate *lam-lam-heh* into a partial Allah ligature by automatically inserting a *shadda* and *superscript alef* over the second *lam*, thereby saving Arabic language speakers a few keystrokes. However, users of other languages dislike this automatic feature, because the *shadda* and *superscript alef* are inserted in words that don't have them.

Currently, the Core Spec (p. 397) includes the following language on *lam-lam-heh*:

> Extra care should be taken not to form the ligature in the absence of the *shadda* and the *superscript alef*, as the sequences <*alef, lam, lam, heh*> and <*alef, lam, lam, shadda, heh*> exist in Persian and other languages with different meanings or pronunciations, where the formation of the ligature would be incorrect and inappropriate.

The following options to address the issue were mentioned:

- Insert a ZWJ to prevent the ligature. This suggestion is contained in a preliminary document "Special Rules for Rendering Allah" by SIL (accessible from https://software.sil.org/scheherazade/allah/). However, it was noted that many users may not have a way to easily enter ZW(N)J on their keyboard. Also, the invisibility of such characters is generally problematic for the general user.
- Use language tagging, though this is not widely supported yet.
- Recommend fonts not form the LLH ligature unless there is a *shadda* on the second *lam*.
- Rely on input methods to prevent automatic ligation

After discussion, we recommend the documentation in “Special Rules for Rendering Allah” from SIL be made into a Unicode Technical Note. The UTN could later be filled out with a fuller discussion of input versus font handling, and discuss the differentiation between ligation and automatic insertion of *shadda* and *superscript alef.* In addition, legacy concerns should be discussed, so implementers know what to change and how to deal with existing documents, so any changes will not end up breaking documents.

Recommendations: The following action is recommended:
Action Item for Lorna Evans: Write a Unicode Technical Note on the ligation of *lam-lam-alef* based on the SIL document “Special Rules for Rendering Allah” based on feedback [Wed Jan 26 10:38:02 CST 2022]. (Reference: section 15 of L2/22-068)

16 Egyptian Hieroglyphs

Action: For UTC discussion and decision

Document: L2/22-063 Comments on Public Review Issues (PRI #442 Unicode 15.0.0 Alpha Review)

Comments: We reviewed feedback from Charlotte Buff (dated Mon Feb 28 11:36:43 CST 2022), who recommended the general category for U+13440 EGYPTIAN HIEROGLYPH MIRROR HORIZONTALLY be changed from Cf (Format) to Mn (Nonspacing_Mark), and changes to proposed Line_Break property and Bidi_Class value with the following rationale:

> U+13440 applies only to the hieroglyph immediately preceding it, with possibly a variation selector for rotation intervening. As such, it behaves like a combining mark. Its line break property value would also need to change from GL (Glue) to CM (Combining Mark) accordingly, and its bidirectional class from L (Left_to_Right) to NSM (Nonspacing_Mark).

Andrew Glass, co-author of the proposal for additional Ancient Egyptian Hieroglyph control characters (L2/21-248), agreed with the change in the general category. The SAH also agreed. It was noted that there are other characters with general category Mn with no ink: Variation Selectors, Mongolian FVSes, Signwriting rotator and filler modifiers, TIFINAGH CONSONANT JOINER, and BRAHMI NUMBER JOINER.

Recommendations: We recommend the UTC make the following dispositions:
SAH-UTC171-R9: Change the General_Category property for U+13440 EGYPTIAN HIEROGLYPH MIRROR HORIZONTALLY from Cf to Mn in Unicode 15.0 (Reference: section 16 of L2/22-068)

SAH-UTC171-R10: Change the Line_Break property for U+13440 EGYPTIAN HIEROGLYPH MIRROR HORIZONTALLY from GL to CM in Unicode 15.0 (Reference: section 16 of L2/22-068)

SAH-UTC171-R11: Change the Bidi_Class for U+13440 EGYPTIAN HIEROGLYPH MIRROR HORIZONTALLY from L to NSM in Unicode 15.0 (Reference: section 16 of L2/22-068)

The following actions are recommended:
Action item for Ken Whistler: Update UnicodeData.txt to change the General_Category property for U+13440 EGYPTIAN HIEROGLYPH MIRROR HORIZONTALLY from Cf to Mn and the Bidi_Class from L to

NSM for Unicode 15.0, based on feedback [Mon Feb 28 11:36:43 CST 2022]. (Reference: section 16 of L2/22-068)

Action Item for Ken Whistler: Change the Line_Break value of U+13440 EGYPTIAN HIEROGLYPH MIRROR HORIZONTALLY from GL to CM in LineBreak.txt for Unicode 15.0, based on feedback [Mon Feb 28 11:36:43 CST 2022]. (Reference: section 16 of L2/22-068)

---

17 Indic

Action: FYI with action to record

Document: L2/22-063 Comments on Public Review Issues (Subject: Order of Indic cantillation marks)

> **Date/Time: Wed Mar 16 13:13:48 CDT 2022**
> **Name: David Corbett**
>
> What should the relative order be between above- and below-base marks with Indic_Syllabic_Category=Cantillation_Mark and post-base marks like visarga? Microsoft's Indic shaper expects such Vedic marks at the end of the cluster, but Microsoft's USE expects them to be mixed in with other marks, meaning they precede post-base marks

Comments: We reviewed the feedback on Indic Cantillation marks and invite research and a proposal that addresses the problem, including the pros and cons of different possible options.

Recommendations: The following action is recommended:
Action Item for Debbie Anderson: Forward the comments in section 17 of L2/22-068 to David Corbett in response to his feedback [Wed Mar 16 13:13:48 CDT 2022].

---

18 Kawi Line breaks and grapheme clusters

Action: For UTC discussion with action to record

Document: L2/22-063 Comments on Public Review Issues (PRI #441 Proposed Update UAX 29 Unicode Text Segmentation)

The following feedback was submitted for PRI #441 Proposed Update UAX 29 Unicode Text Segmentation. Markus Scherer referred the feedback to the Script Ad Hoc for review.

> **Date/Time: Tue Mar 29 00:43:30 CDT 2022**
> **Name: Norbert Lindenberg**
> **Report Type: Public Review Issue**
> **Opt Subject: 441**

> The proposed update for UAX 29 excludes the following Kawi characters from having the Grapheme_Cluster_Break property value SpacingMark:
> U+11F03 KAWI SIGN VISARGA
> U+11F34 KAWI VOWEL SIGN AA
> U+11F35 KAWI VOWEL SIGN ALTERNATE AA
> U+11F41 KAWI SIGN KILLER
> Being excluded from having SpacingMark means that they receive the Grapheme_Cluster_Break property value Other. In consequence, these characters do not combine with other characters into extended grapheme clusters; they always form their own separate grapheme clusters.
> I don't see any reason in the proposal for Kawi, L2/20-284R, or anywhere else why that should be the case. The purpose of grapheme clusters isn't well defined, but one case where the Unicode Standard recommends using them is in emergency line breaking (see UAX 14, section 3, Introduction). If a line break is introduced before a combining mark of a complex script, fonts or rendering systems commonly insert a dotted circle as a base for that mark, which is undesirable.
> The corresponding spacing combining marks in the three most closely related scripts, Javanese, Balinese, and Sundanese, all have the Grapheme_Cluster_Break property value SpacingMark or (in one case, 1B35) Extend. I suggest that Kawi is handled the same way.

Comments: The following points were raised during discussion:

- In general, the recommendation by Norbert Lindenberg to remove the four Kawi characters from the list of exceptions for SpacingMark in UAX #29 looks acceptable. We suggest changing the Line_Break property values for Kawi to indicate the Western style of context analysis for line breaking. However, careful review and investigation by the Properties and Algorithms group is needed.
- The assumption made in an invariant test between lb=SA and exclusion from GCB=SpacingMark needs to be carefully reviewed by the P&A group.

Recommendations: The following action is recommended:
Action Item for the Properties and Algorithms group: Consider comments in section 18 of L2/22-068 based on PRI #441 feedback [Tue Mar 29 00:43:30 CDT 2022].

____

19 Kawi Name Change and Annotation

Action: FYI with action to record

Document: L2/22-056 Review of the code charts of Unicode 15 Alpha (PRI 442) – Marin Silva

In L2/22-056, the following comments were made:

> In order to be consistent with the Myanmar analogues 102C (MYANMAR VOWEL SIGN AA) and 102B (MYANMAR VOWEL SIGN TALL AA), I recommend renaming 11F35 from KAWI VOWEL SIGN ALTERNATE AA to KAWI VOWEL SIGN TALL AA. I also want to mention, that the note stating that the digit two is also used as the letter ro, is missing.

Comments: Because there are different glyph variants for KAWI VOWEL SIGN ALTERNATE AA, a name change to reflect a particular shape is not recommended. Hence, we recommend no change to the current name. An annotation to KAWI DIGIT TWO is acceptable.

Recommendations: The following actions are recommended:
Action Item for Rick McGowan: Relay the comments in section 19 of L2/22-068 to Eduardo Marin Silva, the author of L2/22-056.
Action Item for Ken Whistler and the EdComm: Add an annotation to KAWI DIGIT TWO
"also used as letter ro" for Unicode 15.0, based on L2/22-056. (Reference: section 19 of L2/22-068 and L2/22-056)

---

20 Kawi Review of Code Chart for 15.0

Action: FYI with action to record

Document: L2/22-093 Review of Kawi code chart of Unicode 15 Alpha (PRI 442) -- Waṣkiṭa Kinaṇṭi and Rikza F. Sh.

Comments: We have reviewed the feedback from the authors regarding the Kawi conjoiner and *repha*. Various options were thoroughly discussed regarding the Kawi conjoiner and *repha* (see page 14 of Script Ad Hoc report L2/20-250, document L2/20-283 and page 28 of the Script Ad Hoc recommendations L2/21-016r). We have considered the authors' input but recommend no change. Note that the reasoning for viramas is the same as for Tulu-Tigalari in section 7. However, we welcome the authors to submit a proposal for a combining nukta character.

Recommendations: The following action is recommended:
Action Item for Norbert Lindenberg: Relay the comments in section 20 of L2/22-068 to the authors of L2/22-093.

---

21 Kayah Li

Action: FYI with action to record

Document: L2/22-063 Comments on Public Review Issues (Subject: IndicPositionalCategory.txt)

> **Date/Time: Sun Mar 13 16:30:37 CDT 2022**
> **Name: David Corbett**
>
> The Kayah Li vowel signs U+A926..U+A92A should have Indic_Positional_Category = Top.

Comments: We agree with the feedback from David Corbett. Kayah Li has Indic syllabic category data, but currently no Indic positional category. The combining classes for the nonspacing marks for Kayah Li were set in expectation of a specific order of marks, with vowels above occurring before tone marks below. The requested property change will provide complete Kayah Li data for the Universal Shaping Engine.

Recommendations: The following actions are recommended:
Action Item for Roozbeh Pournader: Set the Indic_Positional_Category to Top for the Kayah Li vowel signs U+A926..U+A92A for Unicode 15.0, based on feedback [Sun Mar 13 16:30:37 CDT 2022] (Reference: section 21 of L2/22-068)
Action Item for Roozbeh Pournader: Add Kayah Li to the list of scripts which contain dependent vowels or similar characters for Indic_Positional_Category in IndicPositionalCategory.txt for Unicode 15.0, based on feedback [Sun Mar 13 16:30:37 CDT 2022] (Reference: section 21 of L2/22-068)

22 Khojki

Action: For UTC discussion with action to record

Documents:
L2/22-083 Khojki Confusable Sequences – Constable
L2/22-056 Review of code charts of Unicode 15 Alpha (PRI 442) -- Marín Silva
L2/22-063 Comments on Public Review Issues (PRI #442 Unicode 15.0.0 Alpha Review)

Comments: We reviewed Marín Silva's feedback in L2/22-056 on Khojki, PRI #442 feedback dated Mon Mar 7 14:37:54 CST 2022 also by Eduardo Marín Silva, and document L2/22-083 by Peter Constable.

The feedback from Eduardo Marín Silva recommended an annotation and informative alias for one character, U+11202 KHOJKI LETTER I, to convey the semantics of the character and because it could be "confusable with 11240 + 1122C". However, closer review by Peter Constable noted that a more general issue is present for Khojki, and a more comprehensive approach is required. Rather than an annotation and informative alias for one character, Constable recommended a "Do Not Use" table be included in the Khojki block introduction including all the Khojki characters where confusability is an issue.

After discussion, the following table should be included (which is modified from the version in L2/22-083):

For	Use	Do Not Use
𑈁	11201	11200 + 1122C
𑈂	11202	11240 + 1122E
𑈃	11203	11206 + 1122C
𑈅	11205	11200 + 11231
𑈇	11207	11200 + 11233 11200 + 1122C + 11231
◌𑈲	11232	1122C + 11230
◌𑈳	11233	1122C + 11231

Recommendations: The following action is recommended:
Action Item for Debbie Anderson and EdComm: Include a Do Not Use table in the Khojki block introduction shown in section 22 of L2/22-068, for a future version of the standard.

23 Latin: Names for Latin letters for Malayalam

Action: FYI with action to record

Document: L2/22-063 Comments on Public Review Issues (PRI #442 Unicode 15.0.0 Alpha Review)

Comments: We reviewed this Unicode 15.0 Alpha Review feedback from Denis Moyogo Jacquerye [Tue Feb 15 05:08:45 CST 2022], suggesting a change to the names of the six new Malayalam transliteration characters located at U+1DF25..U+1DF2A. Currently, all six contain “mid-height” in their names, but the author requests they be changed to “fishhook,” a term that is found in similar characters. (The names were changed from “with left hook” [L2/21-156] to “mid-height” at the January 2022 UTC meeting, in response to feedback that the name of one new character U+1DF27 “LATIN SMALL LETTER N WITH LEFT HOOK” was identical to U+0272.)

Usage of “fishhook” and “hook” is inconsistent in character names. It was noted that names may not necessarily reflect their origin. For example, U+027F LATIN SMALL LETTER REVERSED R WITH FISHHOOK is actually a stretched iota.

In our view, changing the names is not advisable: the change won’t improve identifiability because the term “fishhook”/”hook” in names is already inconsistent. “MID-HEIGHT LEFT HOOK” as a name is sufficient as a descriptor.

Recommendations: The following action is recommended:
Action Item for Debbie Anderson: Relay the comments in section 23 of L2/22-068 to Denis Moyogo Jacquerye, author of 15.0 alpha feedback [Tue Feb 15 05:08:45 CST 2022].

24 Old Turkic

Action: FYI

Document: L2/22-063 Comments on Public Review Issues (Subject: Regarding to “2021-04-21 Application for Adding Letters to Old Turkish Alphabet — Gökbey Uluç”)

Note: The following feedback came in after the Script Ad Hoc had met, but because the nature of the input is “FYI”, it is included here with a recommendation for the UTC to note it.

Date/Time: Sun Apr 10 09:14:28 CDT 2022
Name: Tuğrul Çavdar

Dear Unicode Consorsium,

Regarding to “2021-04-21 Application for Adding Letters to Old Turkish Alphabet — Gökbey Uluç”: https://www.unicode.org/L2/L2021/21081-old-turkish-add.pdf

Old Turkic alphabet does not have letters corresponding to F, H, V, J, C consonants because there was none of these sounds in the era when Old Turkic was used (before A. D. 840). Also “O/U” vowels are written in same letter: 𐰆 , as “Ö/Ü” in same letter: 𐰇 as well. The current letters of Old Turkic table defined in https://unicode.org/charts/PDF/U10C00.pdf are correct.

Gökbey has fabricated new letters for F, H, V, J, C on his own to write today’s Turkish with Old Turkic letters. He also use “10C0A 𐰊 OLD TURKIC LETTER YENISEI AB” letter for “O” vowel, “uş/ush” letter (one of two letters in his proposal) for “Ö” vowel.

He also use
“10C06 𐰆 OLD TURKIC LETTER ORKHON O/U” letter for “U” vowel only, and
“10C07 𐰇 OLD TURKIC LETTER ORKHON OE/UE” letter for “Ü” vowel only.

His fabricated alphabet is http://2.bp.blogspot.com/-RGZcnZW6bos/VISzObbuwpI/AAAAAAAACeM/0mnoucBfr30/s1600/cagdas-turk-damgalari.png (from his blog: http://kokturukce.blogspot.com/2011/04/yeni-damgalar-yeni-yaz-duzeni.htm)

The reason why he has proposed these two letters is to use for “Ö” and “AH” instead of using for “UŞ/USH” and “IÇ/ICH”. So, he plans to use his fabricated alphabet in digital platforms.

There are many variations of Old Turkic letters as can be seen in: http://www.tamga.org/2014/12/farkl-dillerdeki-kitablarda-kokturuk.html

It is impossible to produce codes for all variations. The current letters in Old Turkic table of Unicode.Org are correct and the direction of the current iç/ich: 𐰱 is correct.

For your information.

Yours sincerely,
Assoc. Prof. Tuğrul Çavdar, Ph. D.
Karadeniz Technical University
Trabzon, Turkey

Recommendations: The UTC notes this document, but take no further action.

____

25 Tamil

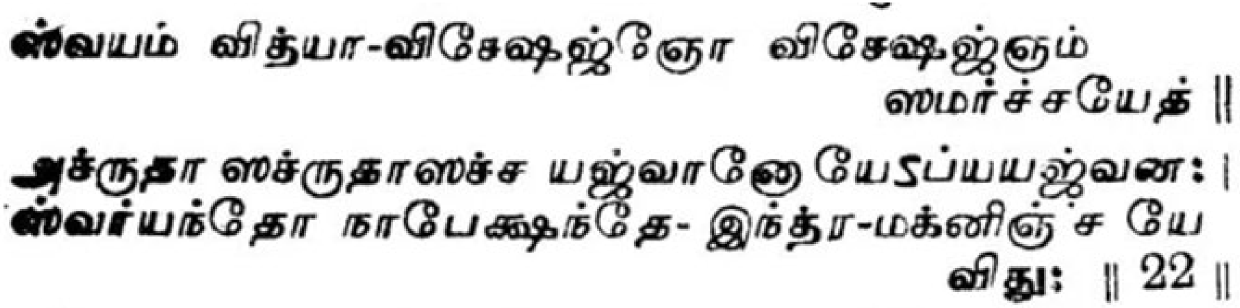
Action: FYI

Document: L2/22-063 Comments on Public Review Issues (Error Report: ScriptExtensions.txt)

Comments: We reviewed the feedback from Elango Cheran dated Mon Feb 7 15:25:04 CST 2022. The author, a speaker of Tamil, questioned U+0964 DEVANAGARI DANDA and U+0965 DEVANAGARI DOUBLE DANDA appearing in Script_Extensions with the Script property value Tamil. The feedback author was not aware of any use-cases with these dandas.

Vinodh Rajan reported that the dandas are used when writing Sanskrit in Tamil (see below). There are also cases of dandas being used outside of Sanskrit.

Below is an example of dandas from a Tamil translation of Sanskrit text (from Sri Lalita Trisati Bhashyam):



ஸ்வயம் வித்யா-விசேஷஜ்ஞோ விசேஷஜ்ஞம்
ஸமர்ச்சயேத் ||
அச்ருதா ஸச்ருதாஸச்ச யஜ்வானோ யேऽப்யயஜ்வன: |
ஸ்வர்யந்தோ நாபேக்ஷந்தே- இந்த்ர-மக்னிஞ் ச யே
விது: || 22 ||

Note: The comments from Vinodh Rajan have already been relayed to the feedback author.

Recommendations: The UTC notes this feedback but takes no action.

---

26 Tibetan marks U+0F35 and U+0F37

Action: FYI with action to record

Document: L2/22-063 Comments on Public Review Issues (Subject: Unhelpful advice…)

> **Date/Time: Thu Mar 3 19:17:44 CST 2022**
> **Name: David Corbett**
>
> Re the Tibetan marks U+0F35 and U+0F37, chapter 13 says “If they are treated as normal combining marks, they can be entered into the text following the vowel signs in a stack”. Should they be treated as normal combining marks? If not, where should they appear in a stack? The standard should clearly specify how to use these code points, and not give such diffident advice.

Comments: We reviewed this feedback.

Andrew West has provided the following input:

> They are not normal combining marks as they apply to the whole stack ("used like underlining" as the code charts note). They should be appended after the last character in a stack, which may be a base consonant, a subjoined consonant, a vowel sign, or some other mark. If two or more of U+0F35 and U+0F37 are appended to the same stack then the behaviour is undefined (the font may render them side-by-side or stacked vertically, or they may end up overlaying each other).

We welcome research and a document on this topic, particularly if a change is deemed necessary to the Core Specification.

Recommendations: The following action is recommended:
Action Item for Debbie Anderson: Forward the comments in section 26 in L2/22-068 to David Corbett in response to his feedback [Thu Mar 3 19:17:44 CST 2022].

---

27 Tibetan encoding of multiple vowels

Action: FYI with action to record

Document: L2/22-063 Comments on Public Review Issues (Subject: How to encode multiple Tibetan vowels at the same height?)

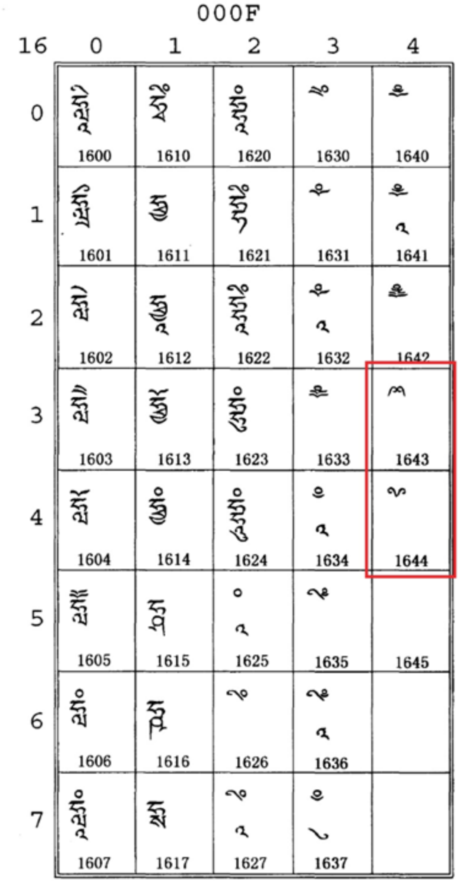
Date/Time: Thu Mar 3 19:44:29 CST 2022
Name: David Corbett

Section 13.4 discusses Tibetan stacks with multiple vowel signs. Usually, when there are multiple vowel signs above the base, they are rendered from bottom to top. In what order should they be encoded when they are rendered side by side? In particular, which of <U+0F68, U+0FA0, U+0F80, U+0F72> and <U+0F68, U+0FA0, U+0F72, U+0F80> is the right encoding for the stack with U+0F80 to the left of U+0F72?

Comments: We reviewed this feedback.

Andrew West has provided the following input:

In this case it is important to note that GB/T 22238-2008 (Tibetan coded character set -- Extension B) defines both sequences of <U+0F80 on left, U+0F72 on right> (F1643) and <U+0F72 on left, U+0F80 on right> (F1644). See attached image of the standard.
It would seem reasonable to me for the encoding order of the signs to match their visual order, i.e. <U+0F80, U+0F72> results in the glyph shown for F1643, and <U+0F72, U+0F80> results in the glyph shown for F1644.



000F 16	0	1	2	3	4
0	1600	1610	1620	1630	1640
1	1601	1611	1621	1631	1641
2	1602	1612	1622	1632	1642
3	1603	1613	1623	1633	1643
4	1604	1614	1624	1634	1644
5	1605	1615	1625	1635	1645
6	1606	1616	1626	1636	
7	1607	1617	1627	1637	

We welcome research and a document on this topic

Recommendations: The following action is recommended:
Action Item for Debbie Anderson: Forward the comments in section 27 in L2/22-068 to David Corbett in response to his feedback [Thu Mar 3 19:44:29 CST 2022].

VIII. OTHER TOPICS

28 African Implementation

Action: FYI with action to record

Document: L2/22-073 Update on implementation status of African scripts - Riley

Comments: We reviewed this summary of the level of implementation for various African scripts.

The following summarizes the discussion:

- This information is valuable for implementers. Rather than being buried in a report on the Unicode website, a better platform might be for the author (and others) to submit such information to ScriptSource, so it can be maintained and widely accessible. (Note: Debbie Anderson has already relayed this recommendation to Charles Riley, author of L2/22-073.)
- Another suggestion was for Debbie Anderson and Ken Whistler to add a link on the Supported Scripts page to ScriptSource.org.

Recommendations: The following action is recommended:
Action Item for Ken Whistler, Debbie Anderson and EdComm: Add a link on the Supported Scripts page to ScriptSource.org (Reference: section 28 of L2/22-068)

29 Bidi Examples

Action: FYI

Document: L2/22-055 Bidi Examples -- Whistler

Comment: We briefly reviewed this document, which presented examples that will help users determine the Bidi_Class for digits and text.

Recommendations: The UTC notes this document, but take no further action.

30 Line Breaking at Orthographic Syllable Boundaries

Action: For UTC discussion with action to record

Documents:
L2/22-080 Line breaking at orthographic syllable boundaries – Lindenberg et al.
L2/22-086 Specification updates for orthographic syllables and line breaking – Lindenberg

Note: The Script Ad Hoc only reviewed L2/22-080. A revised version has been posted as L2/22-080R, which has taken into consideration the comments below. Recommended Core Spec changes that appeared in L2/22-080 are now incorporated in L2/22-086.

Comment: We reviewed L2/22-080. The issues below were noted during discussion of this document. However, the Properties and Algorithms group and others need to fully evaluate the proposed changes.

- The definition of an orthographic syllable on page 5 of L2/22-080 was deemed acceptable. When the text is discussed in the EdComm, recommend a definition for an orthographic syllable to be added to the glossary and link to an updated definition of *aksara*.
- Provide formal long aliases for the proposed new line breaking classes, recognizing that no hyphens are allowed in their names (i.e., Aksara_Prebase); due to stability guarantees, the formal name aliases cannot be changed once published, so care needs to be taken when recommending the names.
- Add examples showing the operation of rule LB28b and other rules involved on actual strings
- How will rules treat clusters that have marks? Add an explanation in cases where marks are present.
- Investigate joiners and nonjoiners for edge cases. If adding special rules for them, be careful around ZWNJ usage in scripts such as Malayalam where it may not break an orthographic syllable. It may be better to be conservative with ZWNJs.
- Investigate quotation marks and other punctuation marks, where dotted circle combinations are immediately preceded and/or succeeded by punctuation marks.
- In the text, call out that Script_Extensions should be considered. For example, Grantha characters that are also used in Tamil.
- Remove support for Myanmar kinzi marks, as the Myanmar script is not a candidate for this style of line breaking.
- Reusing existing line break classes such as ID, where they provide the desired behavior, is preferable to creating new classes.
- For Kawi, show the changes from the line break classes proposed for Unicode 15.

Note that If changes are made to UAX #14 as outlined in this document, script proposal authors should specify if Indic line break style is chosen, and if so, make recommendations on the appropriate line break properties.

For changes to UAX #14, we recommend the UTC members review the proposed changes, and CLDR and ICU implement the proposal and report back on implementation experience and user feedback. The Properties and Algorithm group can report on the feasibility of the changes.

Recommendations: The following actions are recommended:
Action Item for Norbert Lindenberg: Ask CLDR-TC and ICU-TC to consider implementing line breaking at orthographic syllable boundaries, as documented in L2/22-080 (and commented on in section 30 of L2/22-068) and possibly amended after review by the Properties and Algorithms Group, and to report on implementation experience and user feedback.

Action Item for Norbert Lindenberg and the Editorial Committee: Address the text suggestions in L2/22-086 for addition to the Core Spec for Unicode 15.0.