**L2/22-128**

**TO:** UTC
**FROM:** Deborah Anderson, Ken Whistler, Roozbeh Pournader, and Peter Constable[1]
**SUBJECT:** Recommendations to UTC #172 July 2022 on Script Proposals
**DATE:** July 20, 2022

The Script Ad Hoc group met on May 6 and 20, June 10, and July 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 Craig Cornelius, Lorna Evans, Andrew Glass, Liang Hai, Ned Holbrook, Frank van de Kasteelen, Jan Kučera, Norbert Lindenberg, Kamal Mansour, Lawrence Wolf-Sonkin, Michel Suignard, 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. EUROPEAN SCRIPTS

## 1 Cyrillic

### 1a Khanty Cyrillic Letter

**Action: For UTC discussion and decision**

**Recommendations:** We recommend the UTC make the following disposition:
**SAH-UTC172-R1**: Accept U+1C89 CYRILLIC CAPITAL LETTER TJE and U+1C8A CYRILLIC SMALL LETTER TJE, with property values as given in L2/22-119, for a future version of the standard. (Reference: L2/22-119)

The following actions are recommended:
**Action Item** for Ken Whistler: Update the Pipeline to include 1C89 CYRILLIC CAPITAL LETTER TJE and 1C8A CYRILLIC SMALL LETTER TJE. (Reference: L2/22-119)
**Action Item** for Debbie Anderson: Provide Michel Suignard with a font with glyphs for 1C89 CYRILLIC CAPITAL LETTER TJE and 1C8A CYRILLIC SMALL LETTER TJE. (Reference: L2/22-119)

**Document**: L2/22-119 Proposal to encode Cyrillic letter Khanty Tje – Nikita Manulov

**Comments**: We reviewed this request for a case pair used today to write the Khanty language, and specifically the dialects of Kazym, Surgut, and Shuryshkary. The letters designate a voiceless palatal stop, [t']. Further feedback from linguist Tapani Salminen (who proposed other characters for Khanty and other languages in L2/12-052R ) confirmed the letters are used today and needed. However, he recommended dropping "Khanty" from the name since it is a relatively recent pair of letters and could be used for other languages.

The proposed location of the characters in the Cyrillic Extended-C block seems reasonable; and is preferred over Cyrillic Extended-D (which appears in the beta release), since the Extended-D block thus far only has modifier letters and one diacritical mark.

The glyph shape is significantly different from archaic Komi characters U+050E/U+050F CYRILLIC CAPITAL/SMALL LETTER KOMI TJE: Ԏ ԏ (proposed in L2/00-182 = WG2 N2224).

In short, the proposal provides the evidence, with outside confirmation that the case pair is used and needed.

---

### 1b Cyrillic Letters for Bashkir

**Action: FYI**

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

**Document:** L2/22-154 Proposal to encode 18 Cyrillic characters for old Bashkir – Nikita Manulov

**Comments:** We reviewed this proposal for 18 Cyrillic characters for old Bashkir, with examples from primers or charts that originally were published in 1912 and 1928.

The following are comments that came up during discussion:

- It was not clear to the Script Ad Hoc whether these letters were actually adopted in practical usage. To be considered eligible for encoding, a case should be made that these letters were actually used and were not an experimental orthography.
- If the script can be shown to have been used (and was not an experimental orthography), provide examples of the letters in running text.
- On the ISO form (question C.2. on the last page), provide the names of scholars who were contacted, preferably those actively using the characters in their work, and for whom encoding the letters would be important for their work.
- Number the pages.

Debbie Anderson has already forwarded the comments above to the proposal author.

# 2 Latin

## 2a Capital and Small Letter I with Bowl

**Action: FYI**

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

**Documents**:
L2/22-118 Proposal to encode Latin Capital and Small letter I with Bowl -- Nikita Manulov
*Related documents:*
L2/08-404R Proposal to encode four Latin letters for Janalif – Pentzlin and Yevlampiev
L2/10-356 Proposal to encode two Latin letters for Janalif – Pentzlin and Yevlampiev
L2/19-224 Komi Latin letters missing in Unicode (WG2 N5101) -- Rueter
L2/19-261 Proposal to encode Latin letters for the Komi Latin alphabet (supplementing L2/19- 224) – Pentzlin
L2/19-286 SAH recommendations (with comments on L2/19-224)
L2/19-343 SAH recommendations (with comments on L2/19-261)

**Comments:** We reviewed L2/22-118, which is request for a case pair for LATIN LETTER I WITH BOWL. The case pair been proposed several times before, but with the names LATIN CAPITAL CAPITAL/SMALL KOMI BACK I and LATIN CAPITAL/SMALL LETTER YERU.

Because the latest example dates to 1937 and the letters do not appear to be in active use today, we recommend users employ the comparable Cyrillic characters. The UTC recommended that, for the transitional alphabets of the early Soviet era that are no longer used, the Cyrillic characters should be employed rather than to encode Latin look-alikes. See the Script Ad Hoc comments in L2/19-343 and L2/19-286.

Debbie Anderson has already relayed the comments above to the proposal author.

## 2b Latin Letter S with Diagonal Stroke

**Action: For UTC discussion and decision**

**Recommendations:** We recommend the UTC make the following disposition:
**SAH-UTC172-R2:** Accept the following two characters, with properties as given in L2/22-113R, for a future version of the standard:
A7CC LATIN CAPITAL LETTER S WITH DIAGONAL STROKE
A7CD LATIN SMALL LETTER S WITH DIAGONAL STROKE
(Reference: L2/22-113)

The following actions are recommended:
**Action Item** for Ken Whistler: Update the Pipeline to include U+A7CC LATIN CAPITAL LETTER S WITH DIAGONAL STROKE and U+A7CD LATIN SMALL LETTER S WITH DIAGONAL STROKE (Reference: L2/22-113)

**Document:** L2/22-113R Unicode request for two BMP Latin characters – Kirk Miller

**Comments:** We reviewed this document requesting two BMP characters, a case pair used for writing the Luiseño language, a Uto-Aztecan language spoken in California, and in descriptive material for the Cupeño language. Because the characters are used in literacy materials and language-revival program today by the modern-day communities, the author requests they be located in the BMP.

It was noted that the shapes of the proposed case pair for LATIN LETTER S WITH DIAGONAL STROKE are not the same as the glyphs for the LATIN LETTER S WITH OBLIQUE STROKE case pair (U+A7A8 and U+A7A9). We agree these two characters should be atomically encoded with no decomposition. (See Chapter 2.12, sub-section "Overlaid and Attached Diacritics" and 7.9 sub-section "Overlaid Diacritics" of TUS 14.0.)

The evidence is sound and the properties have been provided. We recommend the case pair for encoding.

Note: A font has already been sent by the proposal author to Debbie Anderson (for MIchel Suignard).

## 2c Latin Small Letter Barred Alpha

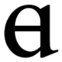
**Action: FYI with action to record**

**Recommendations:** The following actions are recommended:
**Action Item f**or Ken Whistler: Add an annotation to U+AB30 LATIN SMALL LETTER BARRED ALPHA

indicating the character is a Latin alpha and noting that in some fonts the bar may not extend beyond the sides of the alpha. (Reference: L2/22-101)
**Action Item** for Debbie Anderson: Relay comments in section 2c of L2/22-128 to the author of L2/22-101.

**Document**: L2/22-101 Proposal to revise the glyph of LATIN SMALL LETTER BARRED ALPHA – Denis Moyogo Jacquerye

**Comments:** The proposed glyph (below) could be misinterpreted as an *eı* ligature. A revised glyph from the proposal author could be considered, but it should not obscure the identity of the character: that is, it should clearly reflect a Latin alpha (not a Greek alpha) and not be potentially confused with an *eı* ligature. Proposed glyph:



Debbie Anderson has forwarded the document to Michael Everson and Eveline Wandl-Vogt, who both worked on the Teuthonista proposal.

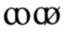
### 2d Latin Small Letter Turned O Open-O/ Latin Small Letter Turned O Open-O with Stroke

**Action: FYI**

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

**Documents**:
L2/22-097 On LATIN SMALL LETTER TURNED O OPEN-O and LATIN SMALL LETTER TURNED O OPEN-O WITH STROKE – Denis Moyogo Jacquerye
L2/22-109 On TURNED O OPEN-O and TURNED O OPEN-O WITH STROKE -- Michael Everson

**Comments**: We reviewed the proposal by Denis Moyogo Jacquerye (L2/22-097), which proposed changing the glyphs for two characters, U+AB43 LATIN SMALL LETTER TURNED O OPEN-O and U+AB44 LATIN SMALL LETTER TURNED O OPEN-O WITH STROKE.

Cf. the following glyph shapes: Current ꭃ ꭄ Proposed: 

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 a Latin *alpha*, not a *turned open o*.

The second document (L2/22-109) contains comments from Michael Everson, who is against the proposed change of glyphs, arguing that what Bremer intended is not important. He adds other comments as well.

It was noted that John Hudson was inclined to support Michael Everson's position – i.e., don't change the glyphs, because the description of the character, its name, and form are all at odds.

Due to conflicting comments, the Script Ad Hoc recommends no action at this time, until additional expert input is received. If the glyphs were to be changed, there would be an outstanding problem on the italic forms, since they would be the same as U+A74F LATIN SMALL LETTER OO and U+A735 LATIN SMALL LETTER AO.

---

### 2e Latin Small Ligature VE

**Action: FYI with action to record**

**Recommendations:** The following action is recommended:
**Action Item** for Debbie Anderson: Forward the comments in section 2e of L2/22-128 to the author of L2/22-106.

**Document**: L2/22-106 Proposal to encode LATIN SMALL LIGATURE VE – Erol Çitci

**Comments**: We reviewed this request for a LATIN SMALL LIGATURE VE, which was created as a Turkish alternative to "&."

No examples of the "ve" ligature in plain text were provided. In our view, this is a ligature and should be handled as such in a font. (Indeed, figure 8 shows the *ve* ligature beside other ligatures in the typeface table.) If the author wishes to represent *ve* in plain text, then the sequence <U+0076, U+200D, U+0065> can be used.

---

## 3 Palaeohispanic

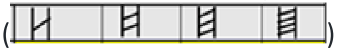
**Action: FYI**

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

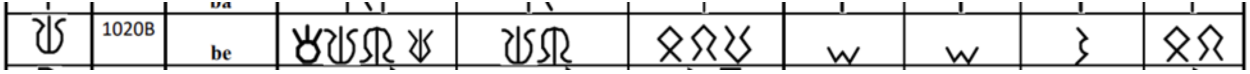
**Documents:**
L2/22-146 Proposal to encode the Northern Palaeohispanic script – Ferrer et al.
L2/22-147 Proposal to encode the Southern Palaeohispanic script – Ferrer et al.
L2/22-148 Re: Palaeohispanic o – Ferrer
L2/22-149 Responses from experts on Palaeohispanic – Anderson

**Comments:** We reviewed the documents on Northern and Southern Palaeohispanic and responses to questions posed by SAH.

The outstanding question primarily involves signs that have different numbers of rungs, and whether they should be separately encoded. Even though a given alphabet may only have 2- or 3-way distinction, the Script Ad Hoc had requested a 3- or 4-way distinction be part of the encoding, so users would be able to select the character based on the graphical element appearing in a text, without requiring additional information.

In Northern Palaeohispanic, the “o” may have one, two, three, or four rungs (). In the Northern Palaeohispanic proposal (section 3, L2/22-146), the experts ask that only two be encoded, a single rung to be used in all inscriptions representing the simplest variant (or if it is the only variant present) or a two-rung version to be used only in scripts with an explicit two-element opposition, indicating the marked variant.

In a recent exchange (L2/22-149) the experts are willing to accept three code points, but point to the case of *be*, asking why *o* is handled differently from *be* (see below):



In our view, the user would need to have additional information in order to select the character (i.e., is this the only variant present or is there a two-element distinction in the text?), as opposed to relying on the grapheme on a given text. The case of *be* is different, because the user does not need to make a linguistically informed choice in the case of *be*.

In Southern Palaeohispanic, the three-bar glyph for *pu-bu* has variants that are 4-, 5-, and 7-bars (section 3 of L2/22-147). It is still not clear whether the 4+ rung characters should be unified with the 3-rung character or not.  Additional information is needed.

# II. MIDDLE EASTERN SCRIPTS

## 4 Arabic

### 4a ARABIC SMALL HIGH LIGATURE ALEF WITH LAM WITH YEH

**Action: For UTC discussion and decision**

**Recommendations:** We recommend the UTC make the following disposition:
**SAH-UTC172-R3:** Accept the glyph change for U+0616 ARABIC SMALL HIGH LIGATURE ALEF WITH LAM WITH YEH as documented in L2/22-104, for Unicode 15.0.

**SAH-UTC172-R4**: Create a formal name alias type correction for U+0616 ARABIC SMALL HIGH LIGATURE ALEF WITH LAM WITH YEH, with the value ARABIC SMALL HIGH LIGATURE ALEF WITH YEH BARREE. (Reference: L2/22-104)

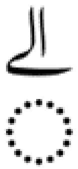
The following actions are recommended:
**Action Item** for Debbie Anderson and the EdComm: Create a glyph erratum for U+0616 ARABIC SMALL HIGH LIGATURE ALEF WITH LAM WITH YEH, based on L2/22-104.  (Reference: section 4a of L2/22-128)
**Action Item** for Lorna Evans: Provide Michel Suignard with a font which includes the corrected glyph for U+0616 ARABIC SMALL HIGH LIGATURE ALEF WITH LAM WITH YEH (Reference: L2/22-104)

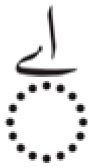
**Action Item** for Ken Whistler: Add a formal name alias type correction for U+0616 ARABIC SMALL HIGH LIGATURE ALEF WITH LAM WITH YEH with the value ARABIC SMALL HIGH LIGATURE ALEF WITH YEH BARREE to NameAliases.txt for 15.0. (Reference: L2/22-104)
**Action Item** for Debbie Anderson: Recommend to L2 that a request for the glyph change for U+0616 ARABIC SMALL HIGH LIGATURE ALEF WITH LAM WITH YEH and a formal name alias type correction for U+0616 ARABIC SMALL HIGH LIGATURE ALEF WITH LAM WITH YEH with the value ARABIC SMALL HIGH LIGATURE ALEF WITH YEH BARREE be added to the DAM ballot comments for ISO/IEC 10646 Amendment 1 based on L2/22-104. (Reference: section 4a of L2/22-128).

**Document:** L2/22-104 Fixing the name and glyph for U+0616 – Pournader

**Comments:** Roozbeh Pournader reviewed an old proposal L2/06-345R, and discovered that the character U+0616 ARABIC SMALL HIGH LIGATURE ALEF WITH LAM WITH YEH was misanalysed due to typographical errors in the typesetting. The character should have been named ARABIC SMALL HIGH LIGATURE ALEF WITH YEH BARREE, with no reference to *lam*.

In our view, this will require a new representative glyph (right, below), a formal name alias with type correction, and a glyph erratum.
Current glyph: Corrected glyph:





## 4b Pegon

**Action: For UTC discussion and decision**

**Recommendations:** We recommend the UTC make the following disposition:
**SAH-UTC172-R5:** Accept the following four characters for a future version of the standard, with property values as given in L2/22-116:
0897 ARABIC PEPET
10EC2 ARABIC LETTER DAL WITH TWO DOTS VERTICALLY BELOW
10EC3 ARABIC LETTER TAH WITH TWO DOTS VERTICALLY BELOW
10EC4 ARABIC LETTER KAF WITH TWO DOTS VERTICALLY BELOW
(Reference: L2/22-116 and section 4b of L2/22-128)

The following actions are recommended:
**Action Item** for Ken Whistler: Correct the annotation to U+06AE ARABIC LETTER KAF WITH THREE DOTS BELOW, referring to U+08B4 instead of U+068A. (Reference: Section 4b of L2/22-128 and L2/22-116)
**Action Item** for Ken Whistler: Update the Pipeline with the four characters
0897 ARABIC PEPET
10EC2 ARABIC LETTER DAL WITH TWO DOTS VERTICALLY BELOW
10EC3 ARABIC LETTER TAH WITH TWO DOTS VERTICALLY BELOW

10EC4 ARABIC LETTER KAF WITH TWO DOTS VERTICALLY BELOW
(Reference: L2/22-116)
**Action Item** for Lorna Evans: Provide a font to Michel Suignard with glyphs for
0897 ARABIC PEPET
10EC2 ARABIC LETTER DAL WITH TWO DOTS VERTICALLY BELOW
10EC3 ARABIC LETTER TAH WITH TWO DOTS VERTICALLY BELOW
10EC4 ARABIC LETTER KAF WITH TWO DOTS VERTICALLY BELOW (Reference: L2/22-116)

**Document**: L2/22-116 Proposal To Encode Four Pegon Characters -- Rikza F. Sh

**Comments:** We reviewed this request for four characters and a request to correct two annotations for the Pegon orthography, the modified Arabic alphabet used to write Modern Javanese, Sundanese and Madurese languages.

The examples support the request for the four characters. The document provides clear evidence that differentiates PEPET from MADDAH ABOVE in the figures. We recommend the code point for PEPET be moved from U+088F to U+0897, however, and the name be changed from ARABIC VOWEL SIGN PEPET to ARABIC PEPET. The Arabic joining type and group will need review by Roozbeh Pournader.

The document also pointed out an error in an annotation for U+06AE ARABIC LETTER KAF WITH THREE DOTS BELOW. Instead of U+068A, U+06AE is an alternative for U+08B4 ARABIC LETTER KAF WITH DOT BELOW.

---

## 4c Three Quranic Arabic Characters

**Action: FYI with action to record**

**Recommendations:** The following action is recommended:
**Action Item** for Debbie Anderson and Roozbeh Pournader: Relay the comments in Section 4c of L2/22-128 to Rikza F. Sh., the author of L2/22-153.

**Document**: L2/22-153 Proposal to Encode Three Quranic Arabic Characters – Rikza F. Sh.

**Comments:** We reviewed this proposal for three Quranic Arabic characters.
The following summarizes the comments on the proposal:

- The case for the first proposed character, RAISED SMALL ALEF, was not deemed convincing to the Script Ad Hoc. In our view, it could be represented by U+0670 ARABIC LETTER SUPERSCRIPT ALEF with a different anchor for the *alef* in the font. Another alternative is to represent it by U+00A0 NO-BREAK SPACE and the *superscript alef*. We recommend removing this character and presenting additional evidence in a separate proposal.
- ARABIC SMALL LOW NOON is reasonable for encoding, but we recommend a different code point. Because it is combining, it should be moved to U+10EFB to be alongside similar characters.

- We agree that ARABIC SMALL YEH BARREE WITH TWO DOTS BELOW is eligible for encoding. However, we recommend the code point be changed to U+10EC5, so it will be with other spacing characters.
- In Section 3, an annotation for U+08AD ARABIC LETTER LOW ALEF, "also used in Quranic orthography" is suggested. We are not convinced by the information provided in the proposal. More information and analysis is needed.
- Section 4 mentions 300 unencoded Arabic characters, and the author recommends a new Arabic Extended-D block be road-mapped. Before such an action is taken, we require a list or chart of the characters.

In sum, we recommend the proposal author revise the proposal, removing RAISED SMALL ALEF and putting it in a separate document, and change the code points for other two characters as noted in the comments. We further recommend the author remove the suggested annotation to U+08AD and create a separate document, providing detailed analysis of the Quranic character and comparison to the original glyph shape of U+08AD in the languages it was originally proposed for (specifically, consider the glyph's height/depth etc.).

# III. SOUTH AND CENTRAL ASIAN SCRIPTS

## 5 Gurung Khema

**Action: For UTC discussion and decision**

**Recommendations:** We recommend the UTC make the following disposition:
**SAH-UTC172-R6**: Accept 58 Gurung Khema characters at U+16100..U+16139, with names and glyphs as given in L2/22-157, and with property values given in L2/22-157 as amended in section 5 of L2/22-128, in a new Gurung Khema block that extends from U+16100..U+1613F for a future version of the standard. (Reference: section 5 of L2/22-128)

The following actions are recommended:
**Action Item** for Ken Whistler: Update the Pipeline to include 58 Gurung Khema characters in a new Gurung Khema block that extends from U+16100..U+1613F. (Reference: L2/22-157 and section 5 of L2/22-128)

**Documents**:
L2/22-157 Proposal to Encode the Gurung Khema in Universal Character Set – Biswajit Mandal
L2/22-132 On the encoding model of the Gurung Khema script -- Eduardo Marín Silva

**Comments:**
***Comments on document from Marín Silva (L2/22-132)***
We reviewed the comments from Eduardo Marín Silva, which were based on an earlier version of the Gurung Khema proposal. A number of suggestions in this document already appeared in the April SAH recommendations L2/22-068:

- We recommended encoding LAILHOMA (anusvara) and MEDIAL RA separately from U+030C COMBINING CARON and U+032D COMBINING CIRCUMFLEX ACCENT BELOW.

- To handle older orthographies, characters can be proposed later (see note above section 3.5 of latest proposal, L2/22-157).
- In our view, the decomposition for II into I – I is not advisable, as the Script Ad Hoc thought it would make the model too complex. The decompositions should, in our opinion, only apply to vertical stacking. (Cf U+030B, which is also atomically encoded.)
- Note section 6 of April 2022 SAH recommendations: "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 requirement about decompositions only containing at most two parts is captured in a Stability Policy (Decomposition_Mapping from 3.0.0+ on https://www.unicode.org/policies/stability_policy.html).
- We also recommended adding a decomposition from SIGN U to SIGN AA + SIGN AA.
- In the latest version of the proposal, L2/22-157, the consonant characters are named with LETTER. (Note that in L2/22-068, "LETTER" appears in the list of properties and the names list, but on p. 8 of L2/22-068, "CONSONANT" appears in the names, which may have caused confusion.)

***Comments on Revised Proposal (L2/22-157)***
In review of the revised proposal, the Script Ad Hoc felt the examples in Section 3.2.2. answered earlier comments on the anusvara, though the current wording ("[anusvara] will sit on top of the base letter and comes after vowel signs") could be improved to clarify encoding order (i.e., "the anusvara is encoded after vowel signs").

11.a. Character Properties
U+1612E GURUNG KHEMA CONSONANT SIGN MEDIAL RA should have bidi class NSM.

11.b. Linebreaking
All entries are missing the general category in the comment.
1612D..161DF can be combined into one entry

11.d. Positional Categories
For Indic_Positional_Category=Left, the code point range should be 1612A..1612B (not 16129..1612A)
For Indic_Positional_Category=Right, the code point range should be 1612C (not 1612B)
For Indic_Positional_Category=Top, code point ranges 1611E..16129 and 1612D, the general category in the comments section should be changed to Mn (not Mc).
For Indic_Positional_Category=Bottom, 1612E..1612F can be combined into one entry
However, these comments are not a blocker for approval of the proposal.

The SAH was in agreement that the proposal could be recommended for approval. A font has already been provided by the proposal author to Debbie Anderson (for Michel Suignard).

## 6 Ol Onal

**Action: FYI**

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

**Document:** L2/22-151  Proposal to encode the Ol Onal script --  Mandal and Kučera
(Note: The Script Ad Hoc only reviewed an earlier version of this proposal.)

**Comments**: We reviewed an early version of L2/22-151, which is a proposal for the Ol Onal script. Ol Onal is used to write Bhumij, a Munda language of the Austroasiatic family that is spoken primarily in the Indian states of Jharkhand, West Bengal and Odisha. The language is commonly written in the Odia, Bengali, Devanagari, and Latin scripts. The script was invented between 1981 and 1992. Although originally a casing script, only uppercase is used today.

The following captures the comments made on the earlier version of the proposal seen by the SAH (but see NOTE below):

- The allocation for Ol Onal be the three columns U+1E5D0..U+1E5FF. This will involve moving Beria, and we recommend a tentative location for Beria be U+1E700.. U+1E7DF.
- Change the General Category for the letter to Lo, rather than Lu. The authors report that lowercase was never used.
- In 3.1.2 LineBreak.txt (page 9), change the line break for OL ONAL SIGN MU and OL ONAL SIGN IKIR from AL to CM.
- Add a new script-specific abbreviation sign, either at every end of the block or after DIGIT NINE.
- Add in information for script metadata, which Debbie Anderson will provide.

NOTE: The authors have already made changes based on the comments above. The revised version of the proposal, which has not yet been reviewed by the Script Ad Hoc, is posted as L2/22-151.

## 7 Tulu-Tigalari

**Action: FYI**

**Recommendations:** The UTC notes this document, but takes no further action

**Document**: L2/22-103 Request to speed up the process of approval to "Proposal to encode Tulu Script in Unicode" – V Sunill Kumar

**Comments**: We briefly noted this document from the Minister for Energy, Kannada & Culture, which mentions that that the Karnataka Tulu Sahitya Academy (KTSA) will be submitting revised documents in the near future for Tulu, and requests the process to encode the Tulu script be sped up.

A note has been sent by Debbie Anderson to the Minister on May 16, 2022, stating that comments from the January 2022 Script Ad Hoc recommendations had been forwarded to KTSA. The comments detail what is needed for a proposal, and that we await the revised document.

*Additional Background*:
The January 2022 Script Ad Hoc recommendations (section 12 of L2/22-023) stated that the Script Ad Hoc had reviewed two proposals: one by the Karnataka Tulu Sahitya Academy (KTSA) for modern Tulu

and one for Tulu-Tigalari by Vaishnavi Murthy and Vinodh Rajan to aid in the digitization of manuscripts. The SAH also reviewed other related documents from the two parties.
The UTC accepted the Tulu-Tigalari script for a future version at its January 2022 meeting, based on the proposal L2/22-031 by Murthy and Rajan. The repertoire approved was "for the archaic Tulu-Tigalari script as seen used predominantly in hand-written manuscripts." The SAH recommendations also contained several comments on the KTSA proposal, with suggestions on how to progress their proposal. The comments were forwarded to the KTSA contact, Dr. Akash Raj, on January 29, 2022.

# IV. SCRIPTS FROM SOUTHEAST ASIA, INDONESIA, AND OCEANIA

## 8 Batak

**Action: FYI with action to record**

**Recommendations:** The following action is recommended:
**Action Item** for Liang Hai and Norbert Lindenberg: Write a document proposing a new Indic Syllabic Category for Batak vowel killers based on their reordering behavior, but consider an orthographic solution (encoding order reform). (Reference: Section 8 of L2/22-128)

**Document**: No document; see description under "Comments," below. (For additional information on Batak, see ScriptSource's compilation of Unicode documents about Batak.)

**Comments**: Batak has two vowel killers: U+1BF2 BATAK PANGOLAT and U+1BF3 BATAK PANONGONAN, which are currently classified Pure_Killer in InSC. This assignment implies [tip] would be represented by <*ta, vowel sign i, pa, killer*>, with two orthographic syllables <*ta, vowel sign i*> and <*pa, killer*>. However, the glyph sequence is expected to be: <*ta, pa, vowel sign i, killer*>, with the *vowel sign i* reordered to attach to *pa*, the base with the killed vowel. However, this is not the expected shaping effect of Pure_Killer and isn't what is expected across an orthographical syllable boundary.

After discussion, the group agreed that Pure_Killer should only be used for viramas that act purely as combining marks and don't have an impact on shaping. Some participants thought the Batak reordering could be seen as conjunct formation; others saw it as something different, but still not "pure."

## 9 Bima

**Action: FYI**

**Recommendations:** The UTC notes this document but takes no action.

**Document:**
L2/22-150 Proposal to Encode Bima Characters in the UCS.pdf --Febri Muhammad Nasrullah
*Related documents:*
L2/16-119 Representing Bima in Unicode -- Pandey
L2/16-075 Proposal to encode VIRAMA signs for Buginese -- Pandey
L2/16-159 Preliminary proposal to encode 'Buginese Extensions' in Unicode -- Pandey

Script Ad Hoc Recommendations:
L2/16-156 section 12, with questions on viramas (May 2016)
L2/16-216, section 11, on Sumbawa, Bima, Lota, and how to integrate all these extensions to Buginese (July 2016)

**Comments**: We reviewed this preliminary proposal to add characters needed to write the Bima language (ISO 639-3:bhp). The proposal includes a chart comparing Bima and Buginese ("Bugis") scripts (on pages 2-4) and demonstrates considerable overlap between the two. The Bima script was used into the 20th century, and is now experiencing a revival, with the script being taught today in primary and secondary schools. The script is still in limited use, however, as modern-day Bimanese communities generally employ the Latin script for everyday use.

In 2016 the Script Ad Hoc reviewed several proposals for extensions to the Buginese script (which may include characters for Bima), and expressed the need for a systematic analysis of the entire set of Buginese extensions in a single document (see L2/16-216, section 11.d). Because a new Bima proposal was submitted, the issue was raised whether the SAH would now consider separate proposals for specific extension sets. Looking at the present Bima proposal, the consensus was that the proposed characters are well justified, and a Bima proposal could progress (after taking into consideration comments below), without requiring a separate unified proposal.

Specific comments on the proposal:

- The proposed consonant letters YA, LA, NCA, NTA, and FA (Section 5.a.) appear well justified. However, the names should be changed to use the language name BIMA before, not after, the consonant, e.g., BUGINESE LETTER BIMA YA. The name change from NYCA (page 3) to NCA (page 6) should be explained.
- For KILLER SIGN, the issue was raised whether the above- and below-base forms should be encoded separately. For transcription of historic documents, we generally assume that authors have control over font, language, and stylistic settings, so separate encoding wouldn't be necessary. However, the proposal also shows variation in the modern transcription in the table in section 4.1 that doesn't just reflect variation in the manuscripts. What caused this variation? Are the modern forms the preferred way to typeset the killer + letter combinations? Or are there preferences per base consonant, or personal preferences of the author? If there are preferences per base consonant, where should the killer be placed for the consonants not shown on the table? Which form(s) is/are taught in schools?
- The proposed REDUPLICATION SIGN (Section 4.2.) is justified, as it differs from JAVANESE PANGRANGKEP both in its shape and its function as a vowel carrier.
- Treating Bima *-e* and *-o* as stylistic variants of their Buginese counterparts is acceptable (for examples, see page 3).
- Provide Information on line breaking and Indic property data.

NOTE: Norbert Lindenberg has already relayed the comments in section 9 of L2/22-128 to the author of the Bima proposal, along with his own comments. The proposal has been revised based on the feedback and been sent to the Script Ad Hoc, but has not yet been reviewed yet.

## 10 New Tai Lue

**Action: FYI with action to record**

**Recommendations:** The following action is recommended:
**Action Item** for Roozbeh Pournader: List all the locations where InPC=Visual_Order_Left, and propose wording for the annotations that would be applicable. (Reference: L2/22-105 and section 10 of L2/22-128)

**Document:** L2/22-105 Annotations for New Tai Lue block – Eiso Chan

**Comments**: We reviewed this request to add the annotation "precede a consonant in visual order" to four New Tai Lue vowels (U+19B5, U+19B6, U+19B7, U+19BA), which will make the name list annotations consistent with Tai Viet (U+AAB5, U+AAB6, U+AAB9, U+AABB, U+AABC), Lao (U+0EC0..U+0EC4), and Thai (U+0E40.U+0E44) which are all Indic_Positional_Category=Visual_Order_Left.

We recommend an Action Item be assigned to Roozbeh Pournader to suggest an annotation that would apply to all applicable occurrences, perhaps "precede a consonant in logical and visual order." It was noted that the standard wording for InPC=Left is "stands to the left of the consonant," but this is not the case for InPC=Visual_Order_Left.

## 11 Yo Lai Tay

**Action: FYI**

**Recommendations:** The UTC notes this document but takes no action.

**Document:** L2/22-152 Preliminary Proposal to encode the Yo Lai Tay Script -- Viet Khoi Nguyen, et al. (Note: The Script Ad Hoc only reviewed an earlier version of this proposal.)

**Comments:** We reviewed an earlier version of this preliminary proposal for the Tai Yo language (ISO 630-3 tyj), a language spoken in Vietnam and Laos. The script is used by some communities in the Nghệ An and Thanh Hóa provinces of Vietnam. Some teaching activities are being conducted with the support of local government and teachers. Structurally, the script is relatively close to Thai, Lao and Tai Viet, with a few exceptions

The following captures the comments made on the last version seen by the SAH (but see NOTE):

- Add code points. From U+1E6C0.. U+1E6FF
- 5.1 Collation: Correct > to < after UE.
- 5.2 Clarify in the text that these are the only two known examples.
- 5.3 Correct the wording: columns run right to left. Also, call out conventions used in the proposal, i.e., inline glyphs are shown horizontally, but in the code chart are vertical.
- Doublecheck names of characters in properties list against names list, i.e., 1xx06 CO in chart but CHO in properties list, cf. also 1xx09, 1xx13, 1xx23, etc.

- 5.6 Remove “(non-standard)”
- Change the line break property for YO LAI TAY SYMBOL MEUANG and YO LAI TAY XAM LAI to AL.

NOTE: The authors already have incorporated changes based on the comments above into new version of the proposal, posted as L2/22-152, but this latest version has not yet been reviewed by the Script Ad Hoc.

# V. EAST ASIAN SCRIPTS

## 12 Chinese Pinyin

**Action: FYI with Action to Record**

**Recommendations:** The following action is recommended:
**Action Item** for Debbie Anderson: Forward the comments in section 12 of L2/22-128 to Eiso Chan in response to his document L2/22-100**.**

**Document:** L2/22-100 Request for consideration to add 4 sequences used for Chinese Pinyin to NamedSequences.txt – Eiso Chan

**Comments:** We reviewed this document asking for four named sequences used in Chinese Pinyin. No strong justification has been provided. The author may consider writing a Unicode Technical Note for the characters and character sequences needed in Pinyin, or could investigate if CLDR has a mechanism for listing them.

## 13 Nüshu

**Action: FYI**

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

**Document:** L2/22-138 Proposal to update the reference glyphs in the Nüshu block and the Ideographic Symbols and Punctuation block – Kushim Jiang

**Comments:** We briefly reviewed this request to update the glyphs for 396 Nüshu characters in the Nüshu block and U+16FE1 NUSHU ITERATION MARK in the Ideographic Symbols and Punctuation block.  Pages 1 and 2 list 22 glyphs that should be corrected.

Since the Script Ad Hoc has no Nüshu experts, we recommended the document be forwarded to others who worked on the Nüshu proposal or have some expertise (i.e., Suzuki, Andrew West, Michael Everson, and the Noto font designer for Nüshu).

The document has been forwarded to Chen Zhuang by Eiso Chan.

Note: Debbie Anderson has already forwarded the document to Suzuki, West, and Everson. It has also been forwarded to Prof. Endo Orie. Liang Hai has forwarded it to the Noto font person for review.

## 14 Rma

**Action: FYI**

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

**Document**: L2/22-130 Preliminary proposal to encode Rma script to UCS – Eiso Chan et al.

**Comments**: We reviewed this preliminary proposal for the Rma script, used to write the Qiang languages (ISO 639-3: cng and qxs), Tibeto-Burman languages spoken in Sichuan province of China. The script is described as a newly developed script, created by Wei Jiuqiao and colleagues. The language is also written in the Latin alphabet.

The following comments were made:

- Provide more details on the history. When was it invented?  How is it being used today? Are there schools teaching the script?  Are there books written by authors other than the creator of the script?
- Is the script's name typically called Rma? Is it also called Qiang?
- In our view, this script does not require Indic properties: it does not have a virama, a repha, or complex features.
- 2.2 Instead of “filler,” we recommend the term “carrier.” Provide explicit examples of the different uses.
- 2.3 Vowel letters and vowel signs. Regarding UGUD and EGED, there was discussion if these should be spacing characters or spacing combining marks.
- The group generally felt spacing combining marks as proposed were acceptable.
- 2.4 Nasalization and rhotacization marks: Provide examples in this section showing nasalization mark. How does it interact with the various letters it occurs with?
- 2.5 Ligature: Provide more information on /ʐme/. What does it mean? Why should it be a separate symbol?
- 2.6 Punctuation: Apparently some Latin punctuation marks are used. Provide a list.
- Tone marks (mentioned on p. 16): Describe the marks and show how they are used.

Note: Norbert Lindenberg has already mailed the author additional extensive comments and questions. Debbie has sent the author the comments above.

## 15 Seal

**Action: FYI with action to record**

**Recommendations:** The following action is recommended:
**Action Item** for Debbie Anderson: Forward the comments in section 15 of L2/22-128 to Selena Wei.

**Documents:**
L2/22-139 TCA and China Feedback on N5133 (WG2 N5133) -- Shouwen Seal Experts
L2/22-140 About the future extension of other versions of Shuowen Seal (WG2 N5188) - TCA
L2/22-141 THX Seal glyph Correction Principles (WG2 N5189) – Selena / TCA
L2/22-142 THX correction glyph summary table (WG2 N5190) –TCA / China
L2/22-143 THX Property Table (14.5MB) (WG2 N5191) –TCA / China
**Comments:** We briefly reviewed the documents on Seal, which were discussed at the June 2022 WG2 meeting.

The approach taken by Taiwan and China has been to rely on a single source for Seal (THX). We are strongly opposed to this approach as it conflicts with agreement at the 2019 Taipei meeting on Seal (WG2 N5119): "Shuowen Seal encoding development process (1) Focus on … THX edition repertory/ordering as base (2) Extend THX repertory by adding additional non-unifiable characters from other major editions."

We note, however, there is no urgency to get Seal into Unicode in the coming year. We also strongly encourage Taiwan and China to involve experts from other countries (such as Japan and US) in their communications and meetings. (Cf. Agreement to coordinate efforts and prevent communication problems, mentioned in WG2 #68 Report on Seal Script Discussion L2/19-245)

# VI. SYMBOLS

## 16 Hexadecimal Digits

**Action: FYI with action to record.**

**Recommendations:** The following action is recommended:
**Action Item** for Debbie Anderson: Relay the comments in section 16 of L2/22-128 to the proposal author of L2/22-131.

**Document**: L2/22-131 Proposal to encode segmented hexadecimal digits from Chinese legacy operation system– Eiso Chan

**Comments**: We reviewed this request to add 6 hexadecimal digits. The following comments were made:

- Clarify what the first four figures demonstrate.
- Show the need today for these characters to be used in public text data interchange (which was also a requirement for the legacy computing symbols).
- Provide the case for why these should be disunified from ASCII. Why can't a font based on ASCII code points be used? Cf. https://www.dafont.com/7led.font

- Are there examples of the digits being mapped to PUA code points?
- If the characters are deemed eligible for encoding, change the answer on the ISO form to YES (8. Can any of the proposed characters be considered a presentation form of an existing character or character sequence?)

## 17 Kodo Incense Linear Patterns

**Action: FYI with action to record**

**Recommendations**: The following actions are recommended:
**Action Item** for Debbie Anderson: Relay comments in section 17 of L2/22-128 to the proposal author of L2/22-117 and the authors of L2/22-155 and L2/22-156.

**Documents**:
L2/22-117 Proposal to encode Kōdō Incense Linear Patterns to UCS – Eiso Chan
L2/22-155 On the Kodo incense patterns -- Marín Silva
L2/22-156 Feedback on L2/22-117 (Proposal to encode Kōdō Incense Linear Patterns) – Wang Yifan

**Comments**:
*Proposal (L2/22-117)*
We reviewed this proposal for 12 characters used to represent Kōdō Incense Linear Patterns, which are used in incense smelling games.

The proposal author should make a clear argument on why the Kōdō Incense Linear Patterns should be encoded as text elements, and why encoding them is needed for public text interchange. Further examples showing the signs being used in-line are needed. If a strong case is made to encode Kōdō Incense Linear Patterns, the encoding model would likely need to be changed, as it relies on 11 control characters and one VS, but no graphic signs.

*Comments on proposal (L2/22-155, L2/22-156)*
We briefly reviewed two documents commenting on the Kōdō Incense Linear Patterns proposal, one from Marín Silva and the other from Wang Yifan. The model proposed by Marin Silva appears more reasonable than that contained in the original document, but the discussion is still premature in our view, since it is not yet clear whether Kōdō linear patterns are text elements needed for interchange.

# VII. PUBLIC REVIEW FEEDBACK AND BETA 15.0 FEEDBACK

## 18 Armenian

**Action: FYI with action to record**

**Recommendations**: The following action is recommended:
**Action Item** for Debbie Anderson: Forward the comments in section 18 in L2/22-128 to David Corbett in response to his feedback [Apr 10 08:59:51 CDT 2022], contained in L2/22-063.

**Document**: L2/22-063 Comments on Public Review Issues (Subject: Armenian left half ring)

> Date/Time: Sun Apr 10 08:59:51 CDT 2022
> Name: David Corbett
> Report Type: Other Document Submission
> Opt Subject: Armenian left half ring
>
> Section 7.6 “Armenian” says “There is no left half ring in Armenian. Unicode character U+0559 is not used. It appears that this character is a duplicate character, which was encoded to represent U+02BB MODIFIER LETTER TURNED COMMA, used in Armenian transliteration. U+02BB is preferred for this purpose.” Via https://en.wiktionary.org/wiki/%D5%99 I found http://www.nayiri.com/imagedBook.jsp?id=1&printPage=10 which shows a left half ring (or turned apostrophe) being used in the Armenian script in a book on Armenian dialects. Should this character be encoded as U+0559 or as U+02BB? The standard should explain which to use in the Armenian script, because the standard is currently wrong or at least misleading**.**

**Comments**: We reviewed this feedback. Further research is needed on this topic, and we invite the feedback author to do the research and write a document recommending specific changes. When doing so, be sure to check with Michel Suignard. Note that U+0559 dates to Unicode 1.1. In addition, a comment about U+02BB appeared in Unicode 4.0 (p. 181).

---

## 19 Brahmi (IndicSyllabicCategory)

**Action: FYI with action to record**

**Recommendations**: The following action is recommended:
**Action Item** for Roozbeh Pournader: Remove the mention of vowel-holders from IndicSyllabicCategory.txt for 15.0, based on feedback from Norbert Lindenberg [April 11 202 17:49:00] in L2/22-063. (Reference: Section 19 of L2/22-128)

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

> Date/Time: Mon Apr 11 17:49:00 CDT 2022
> Name: Norbert Lindenberg
> Report Type: Error Report
> Opt Subject: IndicSyllabicCategory.txt
>
> The file IndicSyllabicCategory.txt has a category Brahmi_Joining_Number, which contains only the Brahmi numbers U+11052..U+11065. The documentation for that category in the same file says "similar to Number in that in can be used as vowel-holders like Consonant_Placeholder, but may also be joined by a Number_Joiner of the same script, e.g. in Brahmi".
> This contradicts the core specification, section 14.1, which says "the numerals U+11052 brahmi number one through U+11065 brahmi number one

> thousand and their ligatures formed with U+1107F brahmi number joiner
> are not used as vowel carriers".

**Comments**: We discussed this feedback. Brahmi distinguishes between decimal and non-decimal numbers. According to Andrew Glass, the attachment of marks to non-decimal numbers in Brahmi is not known, at least based on the work that was done at the time. If additional information contradicting this is uncovered, the Core Spec text and Indic syllabic category can be modified, if needed.

## 20 Chorasmian

**Action: FYI with action to record**

**Recommendations:** The following action is recommended:
**Action Item** for Debbie Anderson: Forward the comments in section 20 in L2/22-128 to David Corbett in response to his feedback [Apr 16 08:57:23 CDT 2022], contained in L2/22-123

**Documents:**
L2/22-123 Comments on Public Review Issues (Subject: Chorasmian number seven)
L2/18-162 Proposal to encode the Chorasmian script (revision 2) -- Pandey

> Date/Time: Sat Apr 16 08:57:23 CDT 2022
> Name: David Corbett
> Report Type: Other Document Submission
> Opt Subject: Chorasmian number seven
>
> How should the Chorasmian number seven on page 47 of L2/18-164R2 be encoded? There is no obvious gap or longer stroke. It is therefore not clear how to use U+10FC5..U+U+10FC8 to represent it, or even whether it can be encoded in Unicode.

**Comments:** We reviewed this feedback. If the feedback author wants to make any change, he is encouraged to do research into the topic and come back with a proposal requesting the specific change(s). The options are discussed on the bottom of page 17 and top of page 18 of the proposal.

## 21 Currency Symbol: Polish ZŁOTY sign

**Action: FYI with action to record**

**Recommendations:** The following action is recommended:
**Action Item** for Debbie Anderson: Thank the author for his feedback [Apr 26 12:15:07 CDT 2022] contained in L2/22-123. (Reference: Section 21 of L2/22-128)

**Document**: L2/22-123 Comments on Public Review Issues (Subject: Feedback on L2/22-092 Proposal to add the currency sign for the POLISH ZŁOTY to the UCS)

> Date/Time: Tue Apr 26 12:15:07 CDT 2022
> Name: Sławomir Osipiuk
> Report Type: Other Document Submission
> Opt Subject: Feedback on L2/22-092 (Proposal to add the currency sign for the POLISH ZŁOTY to the UCS)
>
> I would like to offer additional information which may be of interest to the submitter of L2/22-092. The original proposal omits, to its detriment, that the single-character złoty symbol is also present in the 7-bit character set specified by Polish national standard BN-74/3101-01. As a national standard, this may have more persuasive power for inclusion of this character, and the submitter may want to amend its proposal to include this information.
> Additionally of potential interest, BN-74/3101-01, being a national version of the 7-bit character set conforming to ISO 646, would seem a natural addition to the ISO International Register of Coded Character Sets per ISO 2022 and ISO 2375 (currently managed by the ITSCJ: https://www.itscj-ipsj.jp/english.html ). However, BN-74/3101-01 was never added to the Register for reasons I am not aware of (and the Register itself has not seen any additions since the year 2004). If this character set had been added in the past, then inclusion of the złoty symbol in Unicode/ISO 10646 would have been very likely.

**Comments:** We reviewed this feedback and recommend a note of thanks be sent to the feedback author.

Note: As JTC1/SC2 plans to withdraw ISO 2375 as a standard, that registry of character sets and excape sequences for ISO 2022 is effectively frozen.

---

## 22 Dives Akuru

**Action: FYI with action to record**

**Recommendations:** The following action is recommended:
**Action Item** for Norbert Lindenberg: Add Dives Akuru to a revision of L2/22-080R. (Reference: Section 22 of L2/22-128).
**Action Item** for Norbert Lindenberg and the EdComm: Update the block intro on Dives Akuru to record information on line breaking. (Reference: Section 22 of L2/22-128)

**Document**: L2/22-123 Comments on Public Review Issues (Subject: Dives Akuru line breaking)

> Date/Time: Fri Apr 22 20:34:20 CDT 2022
> Name: David Corbett
> Report Type: Other Document Submission
> Opt Subject: Dives Akuru line breaking
>
> This is feedback on L2/22-080R. Another script with line breaks between

> orthographic syllables is Dives Akuru. L2/18-016R “Proposal to encode Dives
> Akuru in Unicode” says “A word may be broken along orthographic syllables
> at any position at the end of a line.” U+1193F and U+11941 would get lb=AP,
> the letters including the independent vowels would get lb=AK, and U+1193E would get lb= VI.

**Comments:** Anshuman Pandey confirmed that line breaks can occur after any orthographic syllable. As a result, Dives Akuru can be added to L2/22-080R.

---

## 23 Egyptian Hieroglyphs

**Action: FYI with action to record**

**Recommendations:** The following action is recommended:
**Action Item** for Debbie Anderson and Andrew Glass: Take into consideration comments from David Corbett [Apr 27 22:09:17 CDT 2022] in drafting the block intro for 11.4 Egyptian Hieroglyphs, in the “Complex Clusters” section of Egyptian Hieroglyph Format Controls for version 15.0 (Reference: Section 23 of L2/22-128).

**Document**: L2/22-123 Comments on Public Review Issues (Subject: Unclear phrasing re complex quadrats)

> Date/Time: Wed Apr 27 22:09:17 CDT 2022
> Name: David Corbett
> Report Type: Other Document Submission
> Opt Subject: Unclear phrasing re complex quadrats
>
> Section 11.4 says “Sometimes a portion of a graphically complex quadrat could be identified as an atomically encoded character. However, in cases where the use of that atomically encoded character as a component of a quadrat sequence would cause ambiguities or uneven distribution in the structure, then a sequence of simpler hieroglyphs should be used instead, with the appropriate joining controls.” This implies that there exist four contexts for an atomically encoded character.
>
> 1. Not in a quadrat sequence
> 2. Causing ambiguities in a quadrat sequence
> 3. Causing uneven distribution in a quadrat sequence
> 4. In a quadrat sequence without any problems
>
> Does the fourth context really exist? Does it ever make sense to put an atomically encoded character in a quadrat sequence? I don’t think so: I think the quoted passage means that atomically encoded characters in quadrat sequences should always be avoided, because they are always either ambiguous or uneven. However, that is not actually what it says. That sentence should be reworded to something stronger: changing “in cases where” to “because” would fix it.
>
> Alternatively, if the fourth context does exist, it would be helpful for the

standard to provide an example.

**Comments:** We reviewed this feedback and found it could be useful to address in the "Complex Clusters" section of Egyptian Hieroglyph Format Controls (11.4 Egyptian Hieroglyphs) of TUS.

## 24 Kannada (beta feedback)

**Action: FYI with action to record**

**Recommendations:** The following action is recommended:
**Action Item** for Ken Whistler and EdComm: Change the annotation to U+0CF3 KANNADA SIGN COMBINING ANUSVARA ABOVE RIGHT from "Sanskrit" to "Konkani, Havyaka Kannada (Havigannada) and Awadhi" for Unicode 15.0, based on feedback [Tue May 31 14:11:31 CDT 2022] in L2/22-123. (Reference: Section 24 of L2/22-128)

**Document:** L2/22-123 Comments on Public Review Issues (Subject: 453 [SAH])

> Date/Time: Tue May 31 14:11:31 CDT 2022
> Name: Charlotte Buff
> Report Type: Public Review Issue
> Opt Subject: 453 [SAH]
>
> In the code chart for the Kannada block, new character U+0CF3 KANNADA SIGN COMBINING ANUSVARA ABOVE RIGHT is placed under the subhead „Signs used in Sanskrit". However, U+0CF3 is not used for writing Sanskrit, but Konkani, Havigannada, and Awadhi.

**Comments**: We reviewed this feedback on the orthographies using U+0CF3 KANNADA SIGN COMBINING ANUSVARA ABOVE RIGHT. After checking the original proposal L2/21-114, we confirm that the feedback is correct, and the annotation should be changed from "Sanskrit" to "Konkani, Havyaka Kannada (Havigannada) and Awadhi."

## 25 Miao (beta feedback)

**Action: FYI with action to record.**

**Recommendations:** The following action is recommended:
**Action Item** for Debbie Anderson: Relay the comments in section 25 of L2/22-128 to David Corbett, in response to his feedback [Wed Jun 1 14:55:37 CDT 2022] in L2/22-123.

**Document:** L2/22-123 Comments on Public Review Issues (Subject: Indic categories for Miao)

> Date/Time: Wed Jun 1 14:55:37 CDT 2022
> Name: David Corbett
> Report Type: Other Document Submission
> Opt Subject: Indic categories for Miao [SAH]
>
> Miao should have values in IndicSyllabicCategory.txt and IndicPositionalCategory.txt. Although it is not an Indic script, it behaves like one in that all of its characters have ccc=0 and it is not immediately obvious what order they go in. Chapter 18 gives some advice about the order, but it would be helpful to represent it with formal properties.

**Comments**: We reviewed this feedback and checked the final Miao proposal (L2/10-093). Miao is not a Brahmi-based script and as such doesn't follow the Indic model. Hence there is no need for Indic categories. It was also noted by Norbert Lindenberg that the proposed values do not always correspond to USE data files (e.g., Corbett proposed 16F51..16F53 ; Nukta, while the USE data file has Vowel_Dependent for these characters).

Andrew Glass reported that Miao properties are currently defined in override files maintained by Glass on a Microsoft Github project: font-tools/IndicSyllabicCategory-Additional.txt at master · microsoft/font-tools (github.com). Glass opined that ideally these extra properties should be merged with Unicode at some point.

## 26 Modifier Letters (beta feedback)

**Action: (See PAG recs L2/22-124, item UCD2)**

**Recommendation**: (See PAG recommendation UCD2)

**Document**: L2/22-123 Comments on Public Review Issues (Public Review Issue 453)

> Date/Time: Tue May 31 13:46:24 CDT 2022
> Name: Charlotte Buff
> Report Type: Public Review Issue 453
>
> The following new characters in the Cyrillic Extended-D block should be given the Soft_Dotted property for consistency with their base forms:
> U+1E04C MODIFIER LETTER CYRILLIC SMALL BYELORUSSIAN-UKRAINIAN I
> U+1E04D MODIFIER LETTER CYRILLIC SMALL JE
> U+1E068 CYRILLIC SUBSCRIPT SMALL LETTER BYELORUSSIAN-UKRAINIAN I

**Comments**: The Script Ad Hoc agrees with the PAG that the three characters should have the soft-dotted property.

## 27 Nonalphabetic bindus (beta feedback)

**Action: For UTC discussion and decision**

**Recommendations:** We recommend the UTC make the following disposition:
**SAH-UTC172-R7:** Change the following five bindu characters to Alphabetic=Yes by adding them to Other_Alphabetic in PropList.txt:
U+0C04 TELUGU SIGN COMBINING ANUSVARA ABOVE
U+0F82 TIBETAN SIGN NYI ZLA NAA DA
U+0F83 TIBETAN SIGN SNA LDAN
U+11080 KAITHI SIGN CANDRABINDU
U+11081 KAITHI SIGN ANUSVARA
(Reference: Section 27 of L2/22-128)

We recommend the following action:
**Action Item** for Markus Scherer and PAG: Change the following five bindu characters to Alphabetic=Yes by adding them to Other_Alphabetic in PropList.txt:
U+0C04 TELUGU SIGN COMBINING ANUSVARA ABOVE
U+0F82 TIBETAN SIGN NYI ZLA NAA DA
U+0F83 TIBETAN SIGN SNA LDAN
U+11080 KAITHI SIGN CANDRABINDU
U+11081 KAITHI SIGN ANUSVARA
(Reference: Section 27 of L2/22-128)

**Document**: L2/22-123 Comments on Public Review Issues (Subject: Nonalphabetic bindus)

> **Date/Time:** Wed Jun 1 15:50:35 CDT 2022
> **Name:** David Corbett
> **Opt Subject:** Nonalphabetic bindus
>
> Most characters with InSC=Bindu have Alphabetic=Yes. The following exceptions should probably be changed to have Alphabetic=Yes too.
> - U+0C04 TELUGU SIGN COMBINING ANUSVARA ABOVE
> - U+0F82 TIBETAN SIGN NYI ZLA NAA DA
> - U+0F83 TIBETAN SIGN SNA LDAN
> - U+11080 KAITHI SIGN CANDRABINDU
> - U+11081 KAITHI SIGN ANUSVARA

**Comments:** We reviewed this feedback from David Corbett and discussed the background of "Alphabetic": "Alphabetic" indicates the characters are used in words, and is used in the derivation of other properties (see UAX #44: Unicode Character Database and section 4.10 of TUS, on p. 189). Basically, those characters identified as "Other_Alphabetic" in PropList.txt automatically become "Alphabetic."

We agree that the five characters noted by David Corbett should have Alphabetic=Yes, for consistency with other bindus in the standard. They were not originally given the property Alphabetic=Yes, due to various oversights.

## 28 Phags-Pa

**Action: FYI**

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

**Document**: L2/22-123 Comments on Public Review Issues (Subject: TUS Chapter 14, section Phags-Pa)

> Date/Time: Wed Apr 13 16:38:50 CDT 2022
> Name: Asmus
> Report Type: Error Report
> Opt Subject: TUS Chapter 14, section Phags-Pa
>
> (1) I stumbled over a bit of editorial conventions that, while correct,
> were leading astray.
> (2) Looks like there's a loosely worded bit that's not actually correct.
>
> (1) When I just now opened the section at random, it took me a while to mentally switch gears and realize that "letter o" in the passage quoted below was the Phags-pa letter. The conventions are all clear, if you know them, but 'o' is unfortunately not giving any internal hint that it's derived from a transcription. Wish there was something unobtrusive to help guide the reader. (It didn't help that I had "letter o" - the Latin one - on my mind from some other project). Perhaps add the script name here even if redundant??
>
>> The invisible format characters U+200D ZERO WIDTH JOINER (ZWJ) and U+200C ZERO WIDTH NON-JOINER (ZWNJ) may be used to override the expected shaping behavior, in the same way that they do for Mongolian and other scripts (see [†Chapter 23, Special Areas and Format Characters†]). For example, ZWJ may be used to select the initial, medial, or final form of a letter in isolation:
>>
>> <U+200D, U+A861, U+200D> selects the medial form of the letter o
>>
>> <U+200D, U+A861> selects the final form of the letter o
>>
>> <U+A861, U+200D> selects the initial form of the letter o

**[NOTE: Item #1 was discussed by EdComm and has already been taken care of]**

> (2) More importantly there seems to be something possibly misstated here:
>
>> "Conversely, ZWNJ may be used to inhibit expected shaping. For example, the sequence <U+A85E, U+200C, U+A85F, U+200C, U+A860, U+200C, U+A861> selects the isolate forms of the letters i, u, e, and o."

> It should be the case that: the isolate forms for 'i' and 'o' in this example are only selected if they don't join with surrounding characters across the boundaries of the sequence. (There's nothing in the definition of a sequence that prevents it from being embedded in other text).(Can't be sure, but from the table it looks like all vowels are dual joining).
>
> It looks like there's an implicit assumption in the text that the sequence
> is standalone.

**Comments:** We agree with the feedback (on #2) that the text could be clearer. We recommend a sentence be added, stating there is no joining between letters *i, u, e*, and *o*.
Note: Ned Holbrook has already provided Julie Allen with wording to be included in Unicode 15.0.

## 29 Sunuwar

**Action: For UTC discussion and decision.**

**Recommendations:** We recommend the following disposition:

**SAH-UTC172-R8:** Approve the glyphs for Sunuwar as shown in the Sunuwar code chart in L2/22-134 (which corrects and supersedes the glyphs shown in the original proposal and chart in L2/21-157R). (Reference: Section 29 of L2/22-128)

**Documents:**
L2/22-123 Public Review Feedback (Subject: Sunuwar glyph chart error)
L2/22-134 ISO/IEC 10646 repertoire proposal post CDAM1.2 -- Suignard

> Date/Time: Wed Jul 6 10:01:07 CDT 2022
> Name: Deborah Anderson
> Report Type: Other Document Submission
> Opt Subject: Sunuwar chart glyph error
>
> Neil Patel noticed that the glyphs for 11BD2 SUNUWAR LETTER SHYELE
> and 11BDC SUNUWAR LETTER SHYER were swapped in the Sunuwar code chart
> (p. 14 of L2/21-157R). Cf. p. 7 of the proposal, where the glyphs
> are correct.
>
> The correct glyphs appear in Michel Suignard's ISO/IEC 10646
> repertoire proposals post Amd1 (WG2 N5181).
>
> The UTC accepted Sunuwar based on L2/21-157R. I recommend the
> UTC go on record noting the error in the code chart in L2/21-157R,
> noting that the correct glyphs appear in WG2 N5181.

**Comments:** We reviewed the feedback. The UTC accepted Sunuwar based on L2/21-157 in consensus [170-C8] at its January 2022 meeting. We recommend the UTC go on record noting the error in the code chart in L2/21-157R, and refer to the correct glyphs in WG2 N5181 (=L2/22-134).

## 30 Syriac (beta feedback)

**Action: FYI with action to record**

**Recommendations:** The following action is recommended:
**Action Item** for Ben Yang and EdComm to review the feedback [Wed Jun 1 15:37:08 CDT 2022] in L2/22-123 and make any suggestions for defining non-left joining characters for Syriac in section 9.2 and/or 9.3 of TUS. (Reference: Section 30 of L2/22-128)

**Document:** L2/22-123 Comments on Public Review Issues (Subject: Ambiguous rule S2 for Syriac [SAH])

> Date/Time: Wed Jun 1 15:37:08 CDT 2022
> Name: David Corbett
> Report Type: Other Document Submission
> Opt Subject: Ambiguous rule S2 for Syriac [SAH]
>
> Rule S2 applies to an "alaph that has a non-left-joining character to its right" etc., but "non-left-joining character" is not defined. What does it mean?
>
> The most straightforward interpretation is [:Joining_Type!=Left_Joining:]. However, that includes many unsuitable characters like punctuation and spaces. U+0020 has [:Joining_Type!=Left_Joining:], but <U+0020, U+0710> should obviously not select the A_fn glyph of U+0710.
>
> A better definition would be [:Joining_Type!=Left_Joining:]&[:Joining_Type!=Non_Joining:]. There is still one small potential problem: there are two non-joining Syriac letters, U+0861 and U+0866. If a word-final alaph follows one of those, should its glyph be A_fn or X_n? If A_fn, then "non-left-joining character" means something more like [:Joining_Type!=Left_Joining:]&[:L:]. However, I don't know.
>
> In any case, the standard should clearly define "non-left-joining character".

**Comments**: We reviewed this feedback. In our view, clarification is needed in section "Arabic Cursive Joining," section 9.2 of Arabic block of TUS.

## 31 Tai Le

**Action: FYI with action to record**

**Recommendations**: The following action is recommended:
**Action Item** for Roozbeh Pournader to consider adding the five characters listed in public feedback from

Norbert Lindenberg [Jan 17 20:02:43 CST 2022] into his Script Exemplars project. (Reference: Section 31 of L2/22-128)

**Document**: L2/22-018 Comments on Public Review Issues (Subject: ScriptExtensions.txt)

> Date/Time: Mon Jan 17 20:02:43 CST 2022
> Name: Norbert Lindenberg
> Report Type: Error Report
> Opt Subject: ScriptExtensions.txt
>
> The proposal for the Tai Le script, L2/01-369, describes the use of
> five “existing nonspacing diacritics in the UCS” as tone marks in
> an older orthography of the script. Apparently this refers to the
> following characters from the Combining Diacritical Marks block:
> U+0300 COMBINING GRAVE ACCENT
> U+0301 COMBINING ACUTE ACCENT
> U+0307 COMBINING DOT ABOVE
> U+0308 COMBINING DIAERESIS
> U+030C COMBINING CARON
>
> The Script_Extensions property values of these characters in
> Unicode 14.0 do not indicate their use in the Tai Le script.
> They should.

**Comments**: The combining characters listed are typically not listed in Script Extensions because they are used by so many scripts. Roozbeh Pournader has a project on script exemplars, and he may consider adding these characters there. (CLDR has exemplars for languages, but not scripts.)

---

## 32 Todhri

**Action: FYI**

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

**Documents:** L2/22-123 Comments on Public Review Issues (Subject: Issue with precomposed Todhri characters [L2/22-074])

> Date/Time: Wed May 18 02:41:15 CDT 2022
> Name: Charlotte Buff
> Report Type: Other Document Submission
> Opt Subject: Issue with precomposed Todhri characters (L2/22-074)
>
> The recently approved Todhri script (cf. L2/20-188r: Everson, „Proposal for
> encoding the Todhri script in the SMP of the UCS“) includes two letters
> that are formed from a base letter plus a dot diacritic: *U+105C9 TODHRI
> LETTER EI and *U+105E4 TODHRI LETTER U. Per consensus 171-C17, it was

> decided to encode these as precomposed characters with canonical decompositions featuring U+0307 COMBINING DOT ABOVE, as was suggested in L2/22-074 (Pournader, „Todhri encoding options“).
>
> However, this approach is not possible to implement as originally intended. According to section 5.1 of UAX #15, Unicode Normalization Forms:
>
> > »A canonical decomposable character *must* be added to the list of post composition version exclusions when its decomposition mapping is defined to contain at least one character which was already encoded in an earlier version of the Unicode Standard.«
>
> Because COMBINING DOT ABOVE is already encoded, using it as the dot diacritic for Todhri would necessitate adding TODHRI LETTER EI and TODHRI LETTER U to the list of composition exclusions, meaning these two characters could never appear in normalised text. This would make their existence as precomposed characters rather superfluous.

**Comments**: This was discussed in the Properties and Algorithms group and Ken Whistler made a small correction to the text of UAX #15 draft update (below). (No further comment was required by Script Ad Hoc.)

> A canonical decomposable character *must* be added to the list of post composition version exclusions when its decomposition mapping is defined to contain two characters, both of which were already encoded in an earlier version of the Unicode Standard. This criterion is required to maintain normalization stability. Without the composition exclusion, any previously existing sequence of the two characters would change to the newly encoded character in NFC, destabilizing the normalized form of pre-existing text.

---

## 33 Vai

**Action: FYI with action to record**

**Recommendations**: The following action is recommended:
**Action Item** for Debbie Anderson: Forward the comments in section 33 in L2/22-128 to David Corbett in response to his feedback [Mon Mar 28 18:33:50 CDT 2022], contained in L2/22-063.

**Documents:** L2/22-063 Comments on Public Review Issues (Subject: Vai line breaking)
*Related documents:*
L2/22-064 UTC #171 properties feedback & recommendations
L2/22-080 “Line breaking at orthographic syllable boundaries” by Norbert Lindenberg, et al

> Date/Time: Mon Mar 28 18:33:50 CDT 2022
> Name: David Corbett

Report Type: Other Document Submission
Opt Subject: Vai line breaking

This is feedback on L2/22-080. Another script with line breaks between orthographic syllables is Vai. The description in chapter 19 indicates that most Vai letters should have lb=ID, and U+A60B and U+A60C should have lb=BA. The “h-” characters might be ID or BA.

**Comments**: We reviewed this feedback. While Vai has orthographic syllables, it does not have the complexity of orthographic syllables discussed in L2/22-080. The SAH does not have the staff to do the research and propose specific changes to properties. We invite David Corbett or others to do such research and write a proposal with specific changes requested and rationale provided.

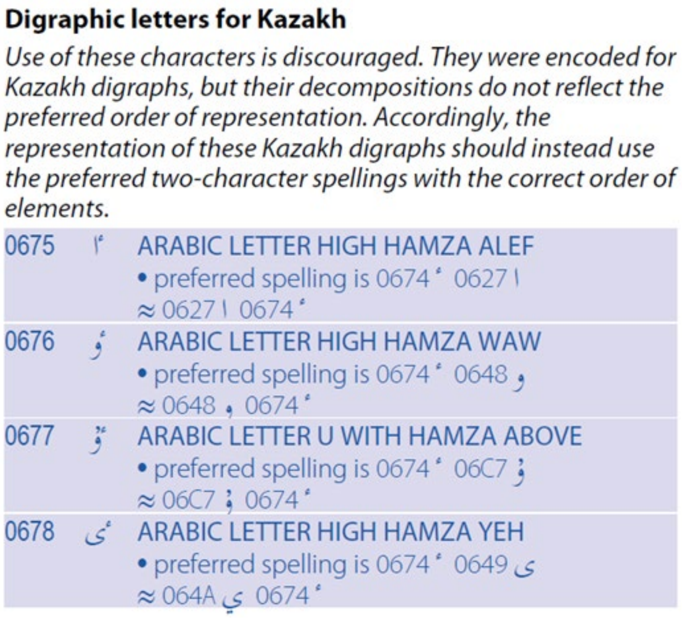
# VIII. NATIONAL BODY FEEDBACK

## 34 China National Body

**Action: FYI with action to record**
**Recommendations:** The following action is recommended:
**Action Item** for Peter Constable and EdComm: Add text to chapter 9, section “High Hamza,” about impact on IDNs of the Kazakh digraphs U+0675..U+0678 for a future version of the standard (Reference: Section 34 of L2/22-128).
**Action Item** for Ken Whistler: In the names list, add “compatibility” in the following sentence under “Digraphic letters for Kazakh” (above U+0675) for 15.0: “These code points were encoded for Kazakh digraphs, but their compatibility decompositions do not reflect the preferred order of representation.” (Reference: Section 34 of L2/22-128)
**Action Item** for Debbie Anderson: Recommend to L2 that the US comments for the ISO/IEC Amendment 1 Draft DAM ballot add the word “compatibility” before “decompositions” in the sentence under “Digraphic letters for Kazakh” (above U+0675). (Reference: Section 34 of L2/22-128)

**Document/Text**:
WG2 N5194 Disposition of Comments on CDAM 1.2 of ISO/IEC 10646 6th edition

Current text in 15.0 and the CD 1.2:



**Digraphic letters for Kazakh**

*Use of these characters is discouraged. They were encoded for Kazakh digraphs, but their decompositions do not reflect the preferred order of representation. Accordingly, the representation of these Kazakh digraphs should instead use the preferred two-character spellings with the correct order of elements.*

| | | |
|---|---|---|
| 0675 | ٵ | ARABIC LETTER HIGH HAMZA ALEF<br>• preferred spelling is 0674 ٴ 0627 ا<br>≈ 0627 ا 0674 ٴ |
| 0676 | ٶ | ARABIC LETTER HIGH HAMZA WAW<br>• preferred spelling is 0674 ٴ 0648 و<br>≈ 0648 و 0674 ٴ |
| 0677 | ٷ | ARABIC LETTER U WITH HAMZA ABOVE<br>• preferred spelling is 0674 ٴ 06C7 ۇ<br>≈ 06C7 ۇ 0674 ٴ |
| 0678 | ٸ | ARABIC LETTER HIGH HAMZA YEH<br>• preferred spelling is 0674 ٴ 0649 ى<br>≈ 064A ي 0674 ٴ |

Comments from China in response to text after “Digraphic letters for Kazakh”:

> “The preferred spelling forms for U+0675 through U+0678 have been included, that are acceptable and suitable. We still argue for removing the sentence “Use of these characters is discouraged.” under the subtitle “Digraphic letters for Kazakh”. These four characters are the proper characters in the Chinese national standards, and have been used widely in China for so many years, which are treated as the single characters in Kazakh alphabet. The word “discouraged” will make the users confused. On the other hand, other pre-combined characters have not been marked as “discouraged”, such as U+00E0 à, so there is no reason to add this for Kazakh-used Arabic letters. The preferred spelling forms are not equal to the UCD mappings. When the users use the new preferred spelling forms in words, they can not search words used with the four characters.”

The final disposition of comments (WG2 N5194) modified the text above U+0675 in the code chart as follows, after having removed the sentence “Use of these characters is discouraged”:

> “These code points were encoded for Kazakh digraphs, but their decompositions do not reflect the preferred order of representation. Accordingly, the representation of these Kazakh digraphs should instead use the preferred two-characters spellings with the correct order of elements.”

**Comments:** We reviewed the comments from China on CDAM 1.2. We recognize the importance of being able to represent the Kazakh letters. Use of U+0675..U+0678, however, will cause implementation problems.

We agree with the Disposition of Comments decision to remove the wording: “Use of these characters is discouraged.” The current text in the Core Spec (p. 394 in chapter 9) already states: “The characters [0675..0678] are not recommended for use.” However, one area *not* mentioned in the Core Spec involves identifiers, since IDNs must be in NFC, but characters not stable under NFKC are prohibited. Adding text to the Core Spec about the impact on IDNs would be an improvement.

There was also discussion about NFKC/NFKD with regard to U+0675..U+0678 with members of PAG.

Note: Subsequent off-line discussion with PAG members resulted in a recommendation to add “compatibility’” in the following sentence for the names list and in the Core Spec:

> “These code points were encoded for Kazakh digraphs, but their compatibility decompositions do not reflect the preferred order of representation.”

A ballot comment on the DAM should be considered.