L2/25-006

UTC #182 properties feedback & recommendations

Markus Scherer & Josh Hadley / Unicode properties & algorithms group, 2025-jan-15

Participants

The following people have contributed to this document:

Markus Scherer (chair), Josh Hadley (vice chair), Elango Cheran, Peter Constable, Mark Davis, Asmus Freytag, Ned Holbook, Robin Leroy, Roozbeh Pournader, Ken Whistler, John Wilcock

1. UCD

1.1 Defective decision 174-C22 on the name of Tolong Siki letters. [#346]

Recommended UTC actions

1. **Consensus**: The names of all Tolong Siki consonants U+11DB6..U+11DD8, including U+11DC5, are the ones without a final A given on pages 8 and 9 of L2/23-024. The inconsistent names given elsewhere in that proposal are not used. See L2/25-006 item 1.1.

PAG input

In Sunnyvale, UTC-174 decided as follows:

> [174-C22] Consensus: Provisionally assign 54 code points U+11DB0..U+11DE9 in a new Tolong Siki block at U+11DB0..U+11DEF, for 54 Tolong Siki characters as described in L2/23-024 and section 6 of L2/23-012.

Section 6 of document L2/23-012 (the SAH report) does not discuss the names of the letters.
The proposal L2/23-024 has three lists of names, all different:

1. pp. 8 sq., in a table (without code points);
2. pp. 10 sq., in UnicodeData.txt lines;
3. p. 13, in a code chart.

See the table below

Ken Whistler reports that the intent of the SAH was to recommend consonant names without an A (P rather that PA, etc.), as on pp. 8 sq.; indeed the proposal, p. 3, states that

> Tolong Siki consonant letters are alphabetic, so they do not possess the inherent *a*.

The names on pp. 10 sq. are almost consistent with those on pp. 8 sq., except for U+11DC5 TOLONG SIKI LETTER C(A).

Character names from L2/23-024, with differences from pp. 8 sq. in **bold**:

pp. 8 sq.	pp. 10 sq.	p. 13
TOLONG SIKI LETTER I	TOLONG SIKI LETTER I	TOLONG SIKI LETTER I
TOLONG SIKI LETTER E	TOLONG SIKI LETTER E	TOLONG SIKI LETTER E
TOLONG SIKI LETTER U	TOLONG SIKI LETTER U	TOLONG SIKI LETTER U
TOLONG SIKI LETTER O	TOLONG SIKI LETTER O	TOLONG SIKI LETTER O
TOLONG SIKI LETTER A	TOLONG SIKI LETTER A	TOLONG SIKI LETTER A
TOLONG SIKI LETTER AA	TOLONG SIKI LETTER AA	TOLONG SIKI LETTER AA
TOLONG SIKI LETTER P	TOLONG SIKI LETTER P	**TOLONG SIKI LETTER PA**
TOLONG SIKI LETTER PH	TOLONG SIKI LETTER PH	**TOLONG SIKI LETTER PHA**
TOLONG SIKI LETTER B	TOLONG SIKI LETTER B	**TOLONG SIKI LETTER BA**
TOLONG SIKI LETTER BH	TOLONG SIKI LETTER BH	**TOLONG SIKI LETTER BHA**
TOLONG SIKI LETTER M	TOLONG SIKI LETTER M	**TOLONG SIKI LETTER MA**

pp. 8 sq.	**pp. 10 sq.**	**p. 13**
TOLONG SIKI LETTER T	TOLONG SIKI LETTER T	**TOLONG SIKI LETTER TA**
TOLONG SIKI LETTER TH	TOLONG SIKI LETTER TH	**TOLONG SIKI LETTER THA**
TOLONG SIKI LETTER D	TOLONG SIKI LETTER D	**TOLONG SIKI LETTER DA**
TOLONG SIKI LETTER DH	TOLONG SIKI LETTER DH	**TOLONG SIKI LETTER DHA**
TOLONG SIKI LETTER N	TOLONG SIKI LETTER N	**TOLONG SIKI LETTER NA**
TOLONG SIKI LETTER TT	TOLONG SIKI LETTER TT	**TOLONG SIKI LETTER TTA**
TOLONG SIKI LETTER TTH	TOLONG SIKI LETTER TTH	**TOLONG SIKI LETTER TTHA**
TOLONG SIKI LETTER DD	TOLONG SIKI LETTER DD	**TOLONG SIKI LETTER DDA**
TOLONG SIKI LETTER DDH	TOLONG SIKI LETTER DDH	**TOLONG SIKI LETTER DDHA**
TOLONG SIKI LETTER NN	TOLONG SIKI LETTER NN	**TOLONG SIKI LETTER NNA**
TOLONG SIKI LETTER C	**TOLONG SIKI LETTER CA**	**TOLONG SIKI LETTER CA**
TOLONG SIKI LETTER CH	TOLONG SIKI LETTER CH	**TOLONG SIKI LETTER CHA**
TOLONG SIKI LETTER J	TOLONG SIKI LETTER J	**TOLONG SIKI LETTER JA**
TOLONG SIKI LETTER JH	TOLONG SIKI LETTER JH	**TOLONG SIKI LETTER JHA**
TOLONG SIKI LETTER NY	TOLONG SIKI LETTER NY	**TOLONG SIKI LETTER NYA**
TOLONG SIKI LETTER K	TOLONG SIKI LETTER K	**TOLONG SIKI LETTER KA**
TOLONG SIKI LETTER KH	TOLONG SIKI LETTER KH	**TOLONG SIKI LETTER KHA**
TOLONG SIKI LETTER G	TOLONG SIKI LETTER G	**TOLONG SIKI LETTER GA**
TOLONG SIKI LETTER GH	TOLONG SIKI LETTER GH	**TOLONG SIKI LETTER GHA**
TOLONG SIKI LETTER NG	TOLONG SIKI LETTER NG	**TOLONG SIKI LETTER NGA**
TOLONG SIKI LETTER Y	TOLONG SIKI LETTER Y	**TOLONG SIKI LETTER YA**
TOLONG SIKI LETTER R	TOLONG SIKI LETTER R	**TOLONG SIKI LETTER RA**
TOLONG SIKI LETTER L	TOLONG SIKI LETTER L	**TOLONG SIKI LETTER LA**
TOLONG SIKI LETTER V	TOLONG SIKI LETTER V	**TOLONG SIKI LETTER VA**
TOLONG SIKI LETTER NNY	TOLONG SIKI LETTER NNY	**TOLONG SIKI LETTER NNYA**

pp. 8 sq.	pp. 10 sq.	p. 13
TOLONG SIKI LETTER S	TOLONG SIKI LETTER S	**TOLONG SIKI LETTER SA**
TOLONG SIKI LETTER H	TOLONG SIKI LETTER H	**TOLONG SIKI LETTER HA**
TOLONG SIKI LETTER X	TOLONG SIKI LETTER X	**TOLONG SIKI LETTER XA**
TOLONG SIKI LETTER RR	TOLONG SIKI LETTER RR	**TOLONG SIKI LETTER RRA**
TOLONG SIKI LETTER RRH	TOLONG SIKI LETTER RRH	**TOLONG SIKI LETTER RRHA**
TOLONG SIKI SIGN SELA	TOLONG SIKI SIGN SELA	TOLONG SIKI SIGN SELA
TOLONG SIKI SIGN HECAKA	TOLONG SIKI SIGN HECAKA	TOLONG SIKI SIGN HECAKA
TOLONG SIKI UNGGA	TOLONG SIKI UNGGA	TOLONG SIKI UNGGA
TOLONG SIKI DIGIT ZERO	TOLONG SIKI DIGIT ZERO	TOLONG SIKI DIGIT ZERO
TOLONG SIKI DIGIT ONE	TOLONG SIKI DIGIT ONE	TOLONG SIKI DIGIT ONE
TOLONG SIKI DIGIT TWO	TOLONG SIKI DIGIT TWO	TOLONG SIKI DIGIT TWO
TOLONG SIKI DIGIT THREE	TOLONG SIKI DIGIT THREE	TOLONG SIKI DIGIT THREE
TOLONG SIKI DIGIT FOUR	TOLONG SIKI DIGIT FOUR	TOLONG SIKI DIGIT FOUR
TOLONG SIKI DIGIT FIVE	TOLONG SIKI DIGIT FIVE	TOLONG SIKI DIGIT FIVE
TOLONG SIKI DIGIT SIX	TOLONG SIKI DIGIT SIX	TOLONG SIKI DIGIT SIX
TOLONG SIKI DIGIT SEVEN	TOLONG SIKI DIGIT SEVEN	TOLONG SIKI DIGIT SEVEN
TOLONG SIKI DIGIT EIGHT	TOLONG SIKI DIGIT EIGHT	TOLONG SIKI DIGIT EIGHT
TOLONG SIKI DIGIT NINE	TOLONG SIKI DIGIT NINE	TOLONG SIKI DIGIT NINE

Background information / discussion

Michel Suignard created the ISO/IEC 10646 7th Edition Committee Draft based on the proposals, and used the names from p. 13.
Robin Leroy and Josh Hadley produced a draft UnicodeData.txt file based on proposals accepted for 17.0, and used the names from pp. 10 sq.
Ken Whistler independently produced a draft UnicodeData.txt file based on the CD and the associated proposals, and used the names from pp. 8 sq.

The discrepancy was noticed when comparing the data files.

1.2 UCD 17 draft Blocks bugs [#356]

Recommended UTC actions

1. **No Action**: These bugs have been fixed in the draft UCD 17 data.

Feedback (verbatim)

Date/Time: Tue Dec 10 01:25:40 CST 2024
ReportID: ID20241210012540
Name: Simon Patrick
Report Type: Error Report
Opt Subject: /Public/draft/UCD/ucd/Blocks.txt

I know that this file is a very early draft for version 17.0 (file is dated 15 November 2024) but you might like to note that (a) I think the new Sidetic block should end at 1095F rather than 1095C and (b) the new Beria Erfe block (16EA0..16EDF) is not in its correct place in code point order: it should come between Medefaidrin (16E40..16E9F) and Miao (16F00..16F9F).

2. Proposed new scripts & characters

PAG members reviewed the following proposals, provided feedback to SAH, and the feedback has been addressed.

No further recommended actions from our side.

- L2/24-139 Proposal to Encode the Jurchen Script and L2/24-140 Proposal to Encode Radicals for the Jurchen Script -- Andrew West, Sun Bojun, Zhōnghuá Zìkù, Michael Everson [SEW #256]
    - A new script propertywise like Tangut, with its own script-specific properties (four rather than just two for Tangut). The radicals are propertywise like the Tangut components; in particular they do not have any of the script-specific properties.
- L2/24-151 Proposal for two geometric shapes for Japanese traditional calendars -- Gen Kojitani [SEW #515]
    - The new characters are generally similar to the already-encoded 六曜 symbols listed in [L2/24-151R, p. 1]; in particular, they have Vertical_Orientation=Upright. The property differences are mostly explained by the existing characters having other usages: the new ones are Math=No, Pattern_Syntax=No, whereas the old ones are Math=Yes, Pattern_Syntax=Yes. Because of their ambiguity, the old ones are lb=Ambiguous, ea=Ideographic, bc=Other_Neutral, whereas the new ones can be lb=Ideographic, ea=Wide, bc=Left_To_Right according to their more restricted usage.
    - Because of the vagaries of roadmapping, the code points used to be Extended_Pictographic. They should no longer be Extended_Pictographic, as they are not (and never will be) emoji.
- L2/24-270 [SEW #591]
    - The properties are similar to those of existing characters in the Cuneiform Numbers and Punctation block, in particular of the existing higher numerals in the 𒁺 series, namely 𒑊 𒑋 𒑌 𒑍 𒑎 (excluding 𒁺 itself, which has different properties on account of its non-numeric usage).
    - The unit of the ASH TIMES n DISH TENU series (𒀸×𒁺) is a normal numeral, instead of being an Other_Letter in the Cuneiform block, as it does not have a non-numeric usage.

3. Collation

3.1 UCA implicit weights for Tangut blocks [#342]

Recommended UTC actions

1. **Consensus**: For Tangut default collation with implicit weights, split Tangut components from Tangut ideographs into separate ranges. Sort Tangut components between Tangut ideographs and Nushu. For Unicode 17.0. See L2/25-006 item 3.1.
2. **Action Item** for Ken Whistler, PAG: For Tangut default collation with implicit weights, split Tangut components from Tangut ideographs into separate ranges. Sort Tangut components between Tangut ideographs and Nushu. For Unicode 17.0. See L2/25-006 item 3.1.

Feedback (verbatim)

Date/Time: Wed Oct 30 07:39:32 CDT 2024
ReportID: ID20241030073932
Name: Andrew West
Report Type: Error Report
Opt Subject: UTS #10 Unicode Collation Algorithm

UTS #10 Unicode Collation Algorithm defines implicit weights for Tangut ideographs and Tangut components (see Table 16 Computing Implicit Weights) with the following formulas:
AAAA = 0xFB00
BBBB = (CP - 0x17000) | 0x8000

This worked OK when there were only a Tangut block and a Tangut Components block, but after the addition of the Tangut Supplement block in Unicode 13.0, the above formulas result in Tangut ideographs in the Tangut Supplement block sorting after all the Tangut components, rather than sorting immediately after the Tangut ideographs in the Tangut block, as would be expected by users. The situation will be even worse after the addition of the Tangut Components Supplement block in a future version of Unicode, when characters in the four Tangut blocks will be sorted in the following order:

Tangut (17000..187FF)
Tangut Components (18800..18AFF)
Tangut Supplement (18D00..18D7F)
Tangut Components Supplement (18D80..18DFF)

The expected default sort order of Tangut ideographs and Tangut components should be:

Tangut (17000..187FF)
Tangut Supplement (18D00..18D7F)
Tangut Components (18800..18AFF)
Tangut Components Supplement (18D80..18DFF)

This could be achieved by separately calculating the implicit weights for Tangut ideographs and Tangut components, as below:

Assigned code points in Block=Tangut OR Tangut_Supplement:
AAAA = 0xFB00
BBBB = (CP - 0x17000) | 0x8000

Assigned code points in Block=Tangut_Components OR Tangut_Components_Supplement
AAAA = 0xFB01
BBBB = (CP - 0x18800) | 0x8000

Assigned code points in Block=Nushu:
AAAA = 0xFB02
BBBB = (CP - 0x1B170) | 0x8000

Assigned code points in Block=Khitan_Small_Script:
AAAA = 0xFB03
BBBB = (CP - 0x18B00) | 0x8000

Background information / discussion

https://www.unicode.org/reports/tr10/#Implicit_Weights

4. Security

4.1 Rare Han characters should not be "recommended" for identifiers [#354]

Recommended UTC actions

1. **Consensus**: Change Identifier_Status for non-common CJK ideographs to Uncommon_Use; that is, CJK ideographs not in RZ-LGR-5. For Unicode 17.0. See L2/25-006 item 4.1.
2. **Action Item** for Asmus Freytag, Markus Scherer, PAG: Change Identifier_Status for non-common CJK ideographs to Uncommon_Use; that is, CJK ideographs not in RZ-LGR-5. For Unicode 17.0. See L2/25-006 item 4.1.

PAG input

Source: Asmus from discussion in PAG meeting

When discussing the need for default assignments for new code points we noted an exception: all new ideographs automatically become "recommended". This seems problematic and different from how all other scripts are treated. Particularly, as few additions are required for "widespread everyday common use" which are our criteria for "recommended".

Problem Statement

There are 97,680 unified CJK ideographs in Unicode, Version 16.0. All of them are part of a recommended script (Hani), but the vast majority of them are unfamiliar to the average user. When used in identifiers, this unfamiliarity is a problem. The goal of identifiers, as explained in RFC 6912, is to serve a “useful mnemonics”, which means that they need to be expressive enough to serve as useful names, but they also need to be distinct enough to allow easy recognition.

This requirement is different from personal or geographical names, where preservation of some exact spelling is important, or for representing general words in a language (which would include its historic forms and precursors). Compared to that, identifiers are deliberately more conservative, largely from a security perspective. At the same time, identifiers are never intended to faithfully represent all words (not even all words in a common / modern subset). Instead, the design point of being "helpful/useful mnemonic" by necessity pairs recognition and relative security with sufficient expressiveness.

> There is a general difficulty in making a hard cutoff for the purpose of delineating "everyday use" Han Ideographs from historical, local or special purpose ideographs. Over the years there have been several attempts at defining a minimal, but sufficient set of characters for modern use. One such effort has been the set of International Ideographs Core [IICORE]; this set accounts for modern, everyday use of Han ideographs in writing the Chinese, Japanese and Korean languages (CJK). [1]

Currently, **all** Han characters are treated as “recommended” for identifiers in UAX#31, which is something that’s probably not helpful, given the goals for identifiers. An optimal recommended subset of ideographs for identifier purposes is much smaller than the full set. To define such a recommended subset, it makes sense to look at industry practice in areas where subsets of ideographs for identifiers have been published. One example of that is registry policies for various levels of domain names.

> In creating the [Maximal Starting Repertoire (MSR)], [the authors] reviewed existing IDN tables for CJK domains and compared them to various subsets, including IICORE, defined in the Unicode Consortium's Unihan database [UAX38]. From this analysis, it appears that the union of certain IDN tables ([JP] and [ZH]) plus the IICORE is most likely to produce a starting set that satisfies the requirement of being larger than the expected final LGR, while at the same time not being overly inclusive.[2]

The various registries for country-code top-level domains (ccTLDs) as well as ICANN for the DNS Root Zone have published repertoire tables. Some of them are subsets that are specific to a given country or language, while others adopt a more regional approach. With the DNS Root Zone being a shared resource, ICANN adopted a superset approach in conjunction with expert teams from the different CJK countries.

This superset of subsets also includes the IICORE repertoire of core ideographs for international use. The actual count comes to 19,842 Unified CJK ideographs[3], including extensions found in actual registry practice.

Recommendation

Revise the Identifier_Status for CJK unified ideographs so that only the ideographs that are identified as relevant / necessary for modern everyday widespread use are listed as "recommended". Ideographs outside this industry supported set should be assigned a status of “Uncommon_Use”. It may not be possible, or useful to accurately assign each ideograph a definite subtype, such as “historic”, “obsolete”, so the suggestion would be to simply mark them as “uncommon_use”. After the change, only the set described in “Data Source” (below) should remain "Recommended".

Data Source

The proposed set of recommended CJK ideographs matches the one documented among other places in Version 5 of the Root Zone LGR, [4] which can be accessed at the location given here, minus the two characters:

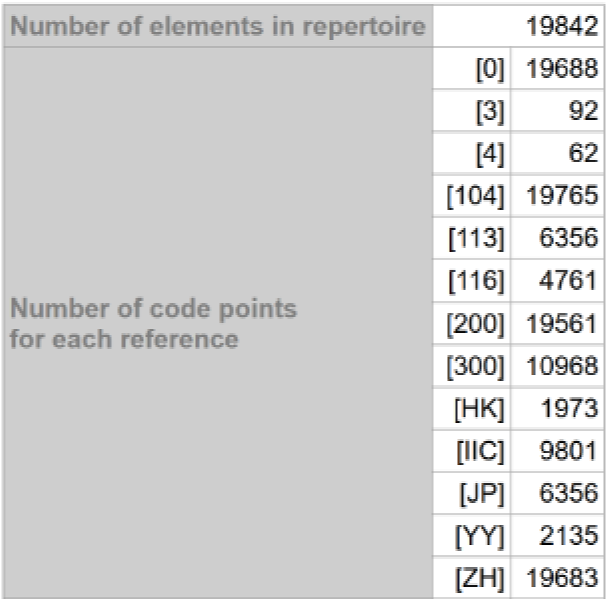
- U+3005
- U+3006

For convenience, the reduced set has been extracted in L2/25-031 with additional source information added to each character. The character collections used in creating this set from a superset of language-specific sets are listed below

Reference	Location
[RZ-LGR-5-Overview]	Integration Panel, “Root Zone Label Generation Rules (RZ LGR-5): Overview and Summary”, 26 May 2022, https://www.icann.org/sites/default/files/lgr/rz-lgr-5-overview-26may22-en.pdf
[RZ-LGR-5]	Integration Panel, “Root Zone Label Generation Rules (RZ-LGR-5)”, 26 May 2022 (XML), https://www.icann.org/sites/default/files/lgr/rz-lgr-5-common-26may22-en.xml, non-normative HTML presentation: https://www.icann.org/sites/default/files/lgr/rz-lgr-5-common-26may22-en.html

Character Collections

The proposed set breaks down as follows:



Number of elements in repertoire		19842
Number of code points for each reference	[0]	19688
	[3]	92
	[4]	62
	[104]	19765
	[113]	6356
	[116]	4761
	[200]	19561
	[300]	10968
	[HK]	1973
	[IIC]	9801
	[JP]	6356
	[YY]	2135
	[ZH]	19683

where the references identify the Unicode versions or character collections cited below. The identification of the references are those from L2/25-031.

Source Collection References

Reference	Source
[0]	Unicode 1.1
[3]	Unicode 3.1
[4]	Unicode 3.2

Reference	Source
[104]	Root Zone Label Generation Rules for the Chinese Script (und-Hani), 26 May 2022 (https://www.icann.org/sites/default/files/lgr/rz-lgr-5-chinese-script-26may22-en.html)
[113]	Root Zone Label Generation Rules for Japanese (und-Jpan), 26 May 2022 (https://www.icann.org/sites/default/files/lgr/rz-lgr-5-japanese-script-26may22-en.html)
[116]	Root Zone Label Generation Rules for Korean (und-Kore), 26 May 2022 (https://www.icann.org/sites/default/files/lgr/rz-lgr-5-korean-script-26may22-en.html)
[200]	CDNC Chinese Characters and Variants Table, https://www.cdnc.asia/file/unicode-1-2.txt,
[300]	Table of General Standard Chinese Characters by China’s State Council, https://www.gov.cn/zwgk/2013-08/19/content_2469793.htm
[HK]	"Hong Kong Supplementary Character Set"
[JP]	IDN Tables for the .jp domain (Japanese) dated 2005-08-30 deposited by Japan Registry Services Co., Ltd. http://www.iana.org/domains/idn-tables/tables/jp_ja-jp_1.2.html
[YY]	List of 2136 jōyō kanji (常用漢字), issued in 2010 by the Japanese Ministry of Education, as listed in: https://en.wikipedia.org/wiki/List_of_j%C5%8Dy%C5%8D_kanji, Visited 2018-02-05
[IIC]	IICORE International Ideographs Core
[ZH]	DotAsia Organisation,“.ASIA ZH IDN Language Table”, 2011-05-04, http://www.iana.org/domains/idn-tables/tables/asia_zh_1.1.txt

Alternatives investigated

In the Unihan Database, there is the UnihanCore2020 set of 20 720 ideographs, which contains 2 853 CJK unified ideographs beyond the MSR's 19 855. In addition, the set contains 70 compatibility characters that are not permissible in identifiers. Overall, the new UnihanCore2020 set has a much larger overlap with the MSR than the much smaller IICORE (9 810 ideographs). However, none of the registries or experts involved in creating the RZ-LGR requested any substantial additions to the MSR. Taken together with the fact that the

Unihan core set includes a number of characters that are not in NFC, it can be questioned whether the 2 853 characters are essential for identifiers, or whether their inclusion was motivated by other concerns.

There are also about 2 056 characters that are in the MSR but not in UnihanCore2020. Because these have been deployed for considerable time in registries for the region, it is not recommended to simply substitute the UnihanCore2020 for the MSR in setting the "recommended" identifier status. (The 70 non-NFC characters would need to be subtracted as well).

Another difference between the sets is that some of the more security conscious identifier systems explicitly support "variant identifiers". These variant definitions are based on similar concepts as the variant definitions for ideographs in Unihan, but they were derived at independently with long-standing practice in registry policies. They do not appear to easily map to the variant definitions in the Unihan database. Using UnihanCore2020 as the "recommended set" or as a subset of the "recommended' set would include 2 853 additional ideographs for which the identifier-relevant variant definitions haven't yet been worked out by anyone.

Given the needs to be conservative in what we recommend for identifiers on the one hand, while also accommodating established practice on the other, would argue for following the Recommendation proposed above. At some later point, it might be possible to increase the Recommended set based on specific evidence that any of the 2 853 ideographs are highly useful for identifiers and do not create security concerns.

[1] Cited from the document describing the development of the Maximal Starting Rerpertoire)

[2] MSR Overview

[3] The source documents cited include two characters U+3005 and U+3006 that are "sc:Hani" but not Unified_Ideograph. They are include in some of the reported summary counts (which include all characters with sc:Hani) but are not relevant here.

[4] Both the Root Zone LGR, or the reference LGRs for the second level, also published by ICANN, contain almost all of the ideographs MSR, except the following 11 characters: {48B5 48BC 48C5 48D3 49D1 4CB3 4D08 5817 974D 9DC0 20B9F}. These were not found to be needed for identifiers and therefore the reduced set actually implemented for the Root Zone is proposed here. The ideograph subset of the MSR can be found here (in highlighted code chart presentation): https://www.icann.org/en/system/files/files/msr-5-han-24jun21-en.pdf

5. Emoji

5.1 Extended_Pictographic assignments for non-Emoji characters [#358]

Recommended UTC actions

1. **Consensus**: Unassign the Extended_Pictographic property from the 672 assigned characters that do not have the Emoji property. For Unicode Version 17.0. See L2/25-006 item 5.1.
2. **Action Item** for Robin Leroy, PAG: In UCD file emoji-data.txt, unassign the Extended_Pictographic property from the 672 assigned characters that do not have the Emoji property. For Unicode Version 17.0. See L2/25-006 item 5.1.

PAG input

From Robin Leroy, PAG, in fulfilment of the following action item:

> [172-A68] Action Item for Mark Davis, PAG: Check Extended_Pictographic values of non-emoji characters for inconsistencies with other similar characters; consider removing ExtPict from non-emoji characters; for a future version of the Unicode Standard. See L2/22-124 item UCD12.

(Feedback quoted in the background section for convenience.)

Mark had noted in L2/22-124 item UCD12 that:

> I would have no real objection to making assigned characters that are not emoji also not be Extended_Pictographic, if (a) we all agree that they can't be emojified (and I think we are there), and (b) we think it is worth the effort (as Buff points out, they don't really hurt anything either).

Re (a), it has become clear that we will not emojify non-emoji anymore. Re (b), while the line breaking algorithm only looks at [\p{Extended_Pictographic}&\p{Cn}], so that Extended_Pictographic assignments do not matter for line breaking of assigned characters, this is not the case of grapheme cluster segmentation; see GB11. We thus incorrectly merge grapheme clusters of non-emoji characters in the presence of ZWJ, which is undesirable (besides being used in ZWJ sequences, ZWJ can be used to request ligatures as an override, and ligatures do not normally merge grapheme clusters).

An invariant test should be added that all assigned Extended_Pictographic characters are Emoji (more specifically, that `\p{Extended_Pictographic}-\p{Cn}=\p{Emoji}-\p{Regional_Indicator}-\p{Emoji_Modifier}-\p{Block=Basic Latin}`).

Background information / discussion

These are the 672 characters assigned in Unicode 16 that are affected by the proposed decision: https://util.unicode.org/UnicodeJsps/list-unicodeset.jsp?a=%5Cp%7BU16%3AExtPict%7D-%5Cp%7BU16%3ACn%7D-%5Cp%7BU16%3AEmoji%7D&g=&i=.

Some 17.0 characters also have incorrect draft data in light of this recommendation. Correcting them does not require a decision.

This proposal addresses both parts of the action item at once: « check Extended_Pictographic values of non-emoji characters for inconsistencies with other similar characters », and « consider removing ExtPict from non-emoji characters ».

There is a historical difference between these two halves. The first half is that some characters have the Extended_Pictographic both incorrectly and unintentionally: their code points had it based on ranges reserved for potential emoji encoding, and the property was not removed when they were encoded. The second half is that some characters have the Extended_Pictographic property (now) incorrectly, but deliberately: these were assigned characters that were deemed potential candidates for emojification, so giving them emoji-like behaviour was sensible for forward compatibility. Emojification is no more, so they should not be Extended_Pictographic anymore.

Original feedback from Charlotte Buff:

```
Date/Time: Fri Jun 24 10:24:49 CDT 2022
Name: Charlotte Buff
Report Type: Public Review Issue
Opt Subject: 453 [PAG]

There are some irregularities in how the Extended_Pictographic property has
been assigned to non-emoji characters, which probably stem from default
values that were never overridden. The following characters are
Extended_Pictographic=True even though none of the other non-emoji
characters within the same blocks share that property:

      U+1F10D..U+1F10F    CIRCLED ZERO WITH SLASH..CIRCLED DOLLAR SIGN WITH OVERLAID BACKSLASH
      U+1F12F                    COPYLEFT SYMBOL
      U+1F16C..U+1F16F    RAISED MR SIGN..CIRCLED HUMAN FIGURE
      U+1F1AD                    MASK WORK SYMBOL
      U+1F260..U+1F265    ROUNDED SYMBOL FOR FU..ROUNDED SYMBOL FOR CAI
      U+1F774..U+1F776    LOT OF FORTUNE..LUNAR ECLIPSE
      U+1F77B..U+1F77F    HAUMEA..ORCUS
      U+1F7D5..U+1F7D9    CIRCLED TRIANGLE..NINE POINTED WHITE STAR
      U+1F8B0..U+1F8B1    ARROW POINTING UPWARDS THEN NORTH WEST..ARROW POINTING RIGHTWARDS THEN
CURVING SOUTH WEST

While there is no real harm to these being Extended_Pictographic, there is
no purpose to it either because none of these characters are ever going to
be emojified and the Extended_Pictographic property has no use outside of
emoji ZWJ sequences.
```

6. Proposed updates

No changes to any UAX, UTS, or UTR at this time. PD-UTS58 was updated recently and renamed to *Unicode Link Detection and Serialization*. We anticipate promoting this to D-UTS #58 during UTC #183