L2/19-186

Unihan Ad Hoc Report on Variant Changes

John H. Jenkins *et al*.

29 April 2019

Summary

The Unihan ad hoc group recommends adding a new field to the Unihan database. The new field will be used to track "spoofing" variants of encoded unified ideographs. The proposed name of the new field is `kSVariant`. Updates to UAX #38 covering the new field described. Further updates to the description of the `kZVariant` field suggested. Description of the method for the initial population of the `kSVariant` field. Description of the method for revising the contents of the `kZVariant` field. Data for both fields to be supplied before UTC #160.

Background

The encoding process used for Unihan involves a three-dimensional model (*TUS* 12.0, pp. 711–712):

The *x*-axis distinguishes *semantics*. 狗 (dog) and 貓 (cat) mean different things, and so occupy different positions on the *x*-axis. This is the basis for the Non-Cognate Rule used in the unification process. The pairs 士/土 and 未/末 both involve forms which would ordinarily be considered unifiable, but as they have distinct meanings (i.e., vary along the *x*-axis), they are encoded as separate characters.

The *y*-axis distinguishes *abstract shape* as defined by the unification rules as codified by the IRG and described in Annex S of ISO/IEC 10646. 貓 and 猫 both mean "cat," but because they have different radicals, they are considered to have distinct abstract shapes and so occupy different positions on the *y*-axis.

The *z*-axis distinguishes *visual style*. It is used to make a distinction between cases such as the three- and four-stroke forms of 艹. Very often there will be a locale-dependent preference for one of these forms or another (such as the right and left variants of 骨), but even then people familiar with the script will recognize them as variants.

One of the weaknesses of the three-dimensional model is that it depends on an "everyman" understanding of the script. It assumes that any two reasonable people will (almost always) agree when two forms are *x*-, *y*-, or *z*-variants. A quarter-century of IRG work has shown that this assumption is perhaps naïve.

The encoding of characters within Unihan is based on the *x*- and *y*-axes. Characters varying on the *x*-axis are encoded separately, as are characters varying on the *y*-axis. Four fields (`kSemanticVariant`, `kSpecializedSemanticVariant`, `kTraditionalVariant`, and `kSimplifiedVariant`) capture different classes of *y*-variation.

In theory, no two *z*-variants of the same ideograph are encoded as separate characters. In practice, this has not happened. The now-defunct Source Separation Rule is responsible for many such encodings; others are the result of simple clerical errors. Two fields, `kZVariant` and `kCompatibilityVariant`, document cases where the same semantic/abstract shape combination has been encoded multiple times. In the case of the `kCompatibilityVariant` field, the normalization process will "fold away" the distinction between the encoded forms. The `kZVariant` field is intended to document the remaining instances where *z*-variants have been separately encoded. Note that no more compatibility ideographs will be encoded. Where they are needed, Ideographic Variation Sequences must be used instead.

The current `kZVariant` data were generated algorithmically a long time ago based on a misunderstanding of the structure of CCCII. It was not proofed at the time and has proven to be of very low quality. It is, in fact, easier to list cases where its data are correct than cases where its data are wrong. These data are long in need of a complete overhaul.

Finally, as a practical issue, there are "confusables," referred to tongue-in-cheek as "*m*-variants" (after Michel Suignard). These are cases where two separately encoded characters are indistinguishable or nearly so in isolation. Many *m*-variants are also *z*-variants (and all *z*-variants are *m*-variants), but there are also cases derived from things such as the near-identity of the "moon" and "meat" radicals (*e.g.*, 朐 and 胸), where even the "man on the street" may not be able to tell them apart.

The *m*-variant issue is closely related to the problem of spoofing. As such, these variants are less whimsically referred to as *s*- or spoofing variants.

Goals

We have two goals here for Unicode 13.0:

1) Replace the data in the `kZVariant` field with something accurate. This would also be a good opportunity to update the syntax along the lines proposed in L2/19-015, if such is desired.
2) Deal with the *s*-variant issue.

The actual data supplied for Unicode 13.0 should focus primarily on useful/important cases. (*I.e.*, start with the URO where there is the greatest practical benefit, then move on to other blocks from there.)

Addition of a New Field to the Unihan Database

Inasmuch as not all *s*-variants are also *z*-variants and because the purpose of tracking *s*-variants (to counter spoofing) is different from the purpose of tracking *z*-variants (to support semantic folding), we recommend that a new field be added to the database. The recommend name of the field is `kSVariant` (for "spoofing"), although names like `kSpoofingVariant`, `kCVariant` ("confusable") and `kVVariant` ("visual") are also options.

A section should be added to UAX #38 to describe the new class of variant. The logical place for this new section would be as §3.7.3, and proposed text would be:

A special class of variant is a spoofing variant (s-variant). S-variants are potentially used in bad faith to direct users to unexpected URIs, evade spam filters, or otherwise deceive end-users. Determining whether or not two characters are s-variants is based entirely on the glyph shape, without regard for semantics. Non-cognate pairs such as 土 and 士 or 未 and 末 are considered s-variants. A common source of s-variants is moon-meat radical confusion, where the two in composed characters look either very similar or identical (e.g., 胸 U+6710 and 胸 U+80CA). Similarly, even if the visual appearance of two radicals is distinct, they may be similar enough that a user might overlook the distinction (e.g., 清 and 清), especially in a spoofing context such as https://清水.org vs.https://清水.org. S-variants also include instances where two highly similar shapes are separately encoded because of source code separation, without regard to other considerations. Such cases include 本 and 本, 刊 and 刊, 幺 and 么.

S-variants might be sufficiently dissimilar in shape that they can be distinguished at large point sizes, or dissimilar in meaning so that they can be distinguished in running text. They might also be visually distinct in one font and not another. These considerations are irrelevant to their status, as such pairs could nonetheless be used to misdirect users (particularly when URIs are displayed at small point sizes).

Because z-variant pairs are by definition either identical or unifiable, they should all be considered s-variants as well. The same consideration holds true for compatibility variants. As such, the `kSVariant` *field only includes s-variants which are not also z-variants or compatibility variants.*

As with some other variant properties, the s-variant property is symmetric (if A is an s-variant of B, then B is an s-variant of A) and transitive (if A is an s-variant of B and B is an s-variant of C, then A is an s-variant of C).

The new field should have a description such as: *"The s-variants (if any) for the character. S-variants are potentially used in spoofing and therefore include all character pairs which look similar, particularly at small point sizes, which are not z-variants. See §3.7.3 for a full description of s-variants."*

The syntax for the `kSVariant` field will be similar to that of the `kZVariant` field. Multiple values are separated by spaces. Each value indicates the encoded variant together with the source. Valid sources include anything corresponding to a field in the Unihan database (*e.g.*, `kHanYu`) or an IRG or UTC document, omitting the standard body prefix, as that can be inferred (*e.g.*, N2310 for an IRG document and 19-2310 for a UTC document). This syntax could be extended to allow WG2 documents as well as IRG and UTC documents to formally used as sources. If, as anticipated, the `kSVariant` and `kZVariant` fields are only extended through UTC actions, only UTC documents need be specified.

There is no need for the letters used to indicate variation type, as with some other variation fields. If a source indicates that two forms are unifiable, then they are *z*-variants. If it indicates that they are different semantically *in any way*, they are *s*-variants. If the source indicates that they have overlapping or identical meanings but are different in abstract shape, then they are some variety of *y*-variant.

As with other variant fields, different authorities may classify a given variant pair differently. The identification of a source makes it possible to distinguish locale-specific differences by indicating different sources.

The overall syntax for the `kSVariant` field would be something like:

```
U\+[23]?[0-9A-F]{4}(<[-A-Za-z0-9]+(,[-A-Za-z0-9]+)*)?
```

The description for the `kZVariant` field will also need to be updated to something like:

The z-variants (if any) for the character. Z-variants are instances where the same abstract shape has been encoded multiple times, either in error or because of source separation. Z-variant pairs also have identical semantics. See §3.7.3 for a full description of z-variants.

Populating the `kSVariant` and `kZVariant` fields

Both the `kSVariant` and `kZVariant` fields will perforce be provisional; that is, the data are considered accurate and useful but not exhaustive.

The current contents of the `kZVariant` field should be jettisoned almost completely. A small number of pairs are either formally documented or have been properly vetted. Annex S, section S.3 of ISO/IEC 10646 has a list of characters separately encoded only because of the source separation rule. This list can be used to flesh out the values for the `kZVariant` field.

Initial data for the `kSVariant` field can be generated from the list in section S.4 of Annex S. This can be further populated using existing phonetic element data in the Unihan database.

The Unihan ad hoc committee will generate (and review) a set of data for the `kSVariant` and `kZVariant` fields for Unicode 13.0, to be submitted to the UTC in time to be discussed at UTC #160.

Future additions to these fields should be disallowed unless the result of action items from the Unicode Technical Committee.