L2/09-080R

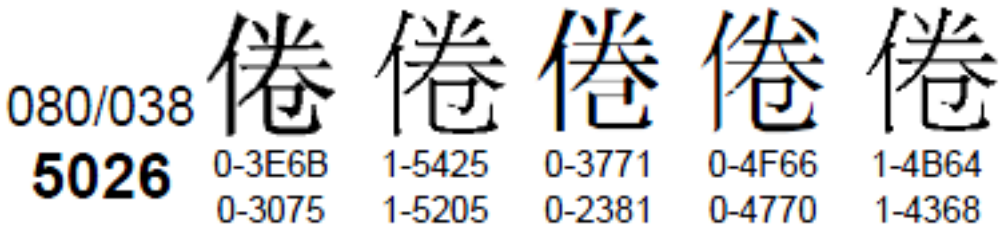
Handling Glyph Shapes for Government Use in WG2/N3530 via Variation Sequences

Source: Unicode Technical Committee and US National Body
Status: Liaison and NB contribution
Authors: Ken Lunde and Eric Muller, Adobe Systems Inc.
Date: February 9, 2009
Action: For consideration by WG2
Distribution: ISO/IEC JTC1/SC2/WG2

§1 Difficulties with compatibility ideographs

Document SC2/WG2/N3525 (L2/08-373) makes the case that CJK compatibility ideographs are not an effective and reliable mechanism to capture glyphic distinctions in plain text. For example, if one wishes to use the shapes 侮 and 侮, simply using the code point sequences <U+4FAE> and <U+FA30> respectively does not *guarantee* the expected rendering: because of the canonical equivalence between the two code points, a process could replace either one by the other.

Furthermore, even in the absence of normalization, the rendering of a given code point is subject to some variation: any shape which can be unified with the representative glyph for the code point is a possible rendering for the code point. This is readily apparent in the multi-column code charts, for example on U+5026:



In a plain text context, any of these unified shapes is an acceptable rendering. Even in the context of a single locale, the preferred shape may vary over time. Indeed, the shape 倦 was preferred for Japanese at the time of the publication of 10646:2003, and the shape 倦 became preferred at a later time. This can be observed in fonts: MS Mincho (XP version) and Kozuka Mincho Pro use the earlier shape while the successors MS Mincho (Vista version) and Kozuka Mincho Pr6N use the later shape.

In summary, the availability of two disctinct code points does not provide control over the selection of a glyph among unifiable shapes. In addition, fi the two code points are canonically equivalent, the distinction between the codes can be erased (by normatlization) at pretty much any point in the processing.

§2 Variation sequences

The same document also explains how variation sequences can be used instead of compatibility characters, and can be an effective mechanism to distinguish unifiable glyph shapes. The sequence <U+4FAE, U+E0100 VARIATION-SELECTOR 17> is an explicit request for the shape 侮 and <U+4FAE, U+E0101 VARIATION-SELECTOR 18> is an explicit request for the shape 侮. Furthermore, the two sequences are not canonically equivalent, so they will not be collapsed by normalization.

The expected renderings of those sequences do not depend on the locale nor on font selection. Even if two fonts produce different shapes for the sequence <U+4FAE>, they should produce 侮 for <U+4FAE, U+E0100> and 侮 for <U+4FAE, U+E0101>. The intent is that <U+4FAE> is used in general, when it is not critical to obtain a specific shape, while the variation sequences are used when the shape is critical.

The two sequences <U+4FAE, U+E0100> and < U+4FAE, U+E0101> are usable in this fashion because they have been registered in the Ideographic Variation Database (IVD), which is available at http://www.unicode.org/ivd and is referenced by 10646.

Of course, the mere inclusion of the mechanism of variation sequences in 10646 is not enough just by itself: implementations (in particular layout engines and fonts) need to implement the mechanism. There are already shipping implementations and more are under development.

The important point is that variation sequences are the only mechanism established by 10646 by which precise control over glyph shapes can be expressed in plain text.

§3 Recasting N3530

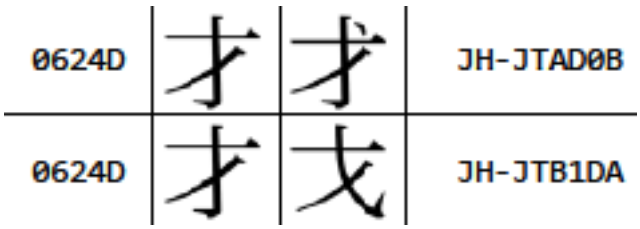
Document SC2/WG2/N3530 (L2/08-371) proposes the encoding of a set of CJK compatibility ideographs, for the explicit purpose of providing control over glyph shapes. For the reasons previously exposed, we believe that variation sequences should be used instead.

The proposal would be transformed in the registration of *n+1* variation sequences whenever *n* compatibility characters are proposed: one for the representative glyph of the unified ideograph and one for the representative glyph of each compatibility character.

For example, the original proposal is to encode a compatibility ideograph for U+5026:

This should instead be a request for the registration of two variation sequences, one for the glyph 倦, and one for the glyph 倦. Then <U+5026> alone can be rendered by either of the two shapes (as it can be today) and each of the two variation sequences can be used to get precisely one of the two shapes.

Similarly, the original proposal is to encode two compatibility ideographs for U+624D:



This should instead be a request for the registration of three variation sequences, one for 才, one for 才, and one for 戈. Then <U+624D> alone can be rendered by any of the three shapes (as it can be today) and each of the three variation sequences can be used to get precisely one of the three shapes.

In addition, it is likely that the set of compatibility ideographs proposed in N3530 is to complement (a subset of) the existing compatibility ideographs. That existing subset should also be included in the registration application.

§4 The process of registration

The process of registration is rather straightforward. If we ignore for the moment the sequences already present in the IVD, the data supporting the registration request would be two files.

The first file, IVD_Sequences.txt would contain one line per variation sequence:

```
...
5026; N3530; J0-3771
5026; N3530; JH-JA2381
...
624D; N3530; J0-3A4D
624D; N3530; JH-JTAD0B
624D; N3530; JH-JTB1DA
...
```

The first field `5026` or `624D` is the code point of the unified CJK ideograph on which the variation sequence is based. The second field `N3530` is some identifier selected by the proposer that designates the collection to which the sequence belong (here we used `N3530`, but this is entirely arbitrary). The third field is a source identifier selected by the proposer; here we used the source_id of N3530 and the source (in the 10646 sense) of the ideograph.

The second file, IVD_Collections.txt, would contain a single line describing the `N3530` collection and would be of the form:

```
N3530; J.+ ; http://example.com/N3530
```

where the first field `N3530` is the identifier of the collection. The second field `J.+` is a regular expression matching the source identifiers in IVD_Sequences.txt. The third field is a URL which points to a document describing the intent of the collection, maintained by the registrant.

Additionally, the registrant is encouraged to provide a font to the registrar, solely for the purpose of creating a code chart (see http://www.unicode.org/ivd/data/2007-12-14/IVD_Charts.pdf for the current code chart).

Following the reception of the registration request, a 90 day review period would start. The purpose is mostly to ensure that the proposed glyph shapes are indeed unifiable. Assuming that no problems are uncovered, the registrar would then assign specific variation sequences to the requested sequences. As of this writing, the sequence <U+624D, U+E0100> is already registered, so the three requested sequences would likely be assigned to <U+624D, U+E0101>, <U+624D, U+E0102>, and <U+624D, U+E0103>. The resulting sequences would then be included in a new version of the Ideographic Variation Database (IVD). Formally, an amendment of 10646 should refer to this new version of the IVD, but in practice the sequences would be usable immediately. As a side benefit, this process is much faster than the encoding of new characters. It should also be noted that for a collection sponsored by ISO/IEC JTC1/SC2, there are no registration fees.

§5 Leveraging the AJ1 collection?

As of this writing, the current version of the IVD (2007-12-14) contains a single collection, registered by Adobe Systems, to accomodate the Adobe-Japan1 (AJ1) glyph complement. The current sequences in that collection cover the AJ1 glyph complement up to supplement 6. The primary target of this collection is desktop publishing applications and electronic documents as used in commercial and government applications.

A comparison of the glyphs proposed in N3530 and of the glyphs in the AJ1 collection shows a certain degree of overlap. For example, the two glyphs for U+5026 are already present, with the sequences <U+5026, U+E0100> for 倦 and <U+5026, U+E0101> for 倦; one of the three glyphs for U+624D is already present, with the sequence <U+624D, U+E0101> for 才. Overall, about 1/3 of the first 200 glyphs proposed in N3530 are present in AJ1.

As a consequence, there are three ways in which to handle the glyphs of N3530:

1. completely ignore the AJ1 collection. As described in the previous section, this would lead to the registration of two new sequences for U+5026, in addition to the two existing sequences, and to the registration of three new sequences for U+624D, in addition to the existing AJ1 sequences. The resulting IVD chart would look like:

5026	倦	倦	倦	倦
	AJ1 E0100	AJ1 E0101	N3530 E0102	N3530 E0103
624D	才	才	才	戈
	AJ1 E0100	N3530 E0101	N3530 E0102	N3530 E0103

2. create a collection that complements the AJ1 collection: This would lead to no registration for U+5026 (the AJ1 sequences would be used) and to the registration of two new sequences for U+624D, in the `N3530` collection, to be used in conjunction with the existing AJ1 sequence augment the AJ1 collection. The resulting IVD chart would look like:

5026	倦	倦	
	AJ1 E0100	AJ1 E0101	
624D	才	才	戈
	AJ1 E0100	N3530 E0101	N3530 E0102

3. similar to the previous option, but the new sequences would be registered in the AJ1 collection rather than in a new `N3530` collection. The resulting IVD chart would look like:

5026	倦	倦	
	AJ1 E0100	AJ1 E0101	
624D	才	才	戈
	AJ1 E0100	AJ1 E0101	AJ1 E0102

The main advantage of options 2 and 3 is that they would make the IVD much simpler for the end users, as a given glyph shape would occur only once. With option 1, users will have to ask themselves "I want to obtain precisely the glyph 倦; should I use the AJ1 sequence or the N3530 sequence?" With option 2 and 3, a given glyph shape would occur only once, thus removing the confusion.

Another factor to take into account is that the domain of application and the overall magnitude (number of sequences) of AJ1 and N3530 are very similar. Both are meant to be used in modern texts (including proper names), and to be relevant for a large part of the Japanese user community, which make the possibility of confusion more troubling. By contrast, a user would have much less difficulty choosing between a collection for modern use and a collection for scholarly use.

Because of those considerations, we recommend investigating options 2 and 3. The choice between the two is of limited technical consequence. There is no danger of AJ1 sequence changing or disappearing from the IVD, as once a sequence is registered, it remains permanently in the IVD (just like a character is never removed nor moved in 10646). In any case, the choice between the three options is entirely in the hands of the registrant.