L2/02-002

To:	UTC
From:	Mark Davis
Re:	Variation Selector Semantics
Date:	2002-01-06

There has been a fair amount of discussion on the variation selector semantics on the mailing list. I think Peter (in the attached mail) has summed it up nicely, and I started out agreeing with him. As I thought about it more, I was, however, led to a somewhat different conclusion.

Our goal is to make very sure that we make the semantics of VS's absolutely clear before we introduce the characters. Here is the way I view the situation:

- Any character has a very large set S of possible glyphs* that can represent it.
- A particular <character, VSx> combination specified in StandardizedVariants.html simply restricts the glyphs to a subset S1 based upon some feature, e.g. "loopy" glyphs.
- It does *not* restrict it to a single glyph, since there is *still* possible variation in the shape (stem-width, aspect, x-height, serifs, etc.).
- Nor does the existence of the <character, VSx> specification in StandardizedVariants.html have any affect on S whatsoever.

In order to have a determinate, disjoint set of glyphs S2 associated with the character, one would have to have an additional <character, VSy> combination specified in StandardizedVariants.html, such as "non-loopy" glyphs. That is, if we wanted to enable mathematicians to reliably distinguish two different shapes for loopy vs non-loopy PHI, we would have to have two different VSn's, one for loopy and one for not.

Where I started disagreeing with Peter is when he said that every time we add a <character, VSy> combination that we have to add *all* of them. That is impossible, since we cannot foresee what peculiar combination of features that mathematicians or CJK users will find distinctive. We just have to recognize the above facts, and if we want disjoint alternatives, then we have to specify them as we need them. For example, suppose we have

> <not-equals, VS1> => glyph must have angled slash

There is still absolutely nothing to prevent a font from representing a simple not-equals with either an angled slash or a non-angled slash. If we want to ensure that people can make that distinction, then we need to add, for some y

> <not-equals, VSy> => glyph must not have angled slash

I disagree with Asmus that y must be 2; it could be anything. We might in the meantime have specified:

> <not-equals, VS2> => glyph must be bold
> <not-equals, VS3> => glyph must be non-bold

and thus end up with:

> <not-equals, VS4> => glyph must not have angled slash

Note that if we did need such a distinction, we might be in the situation of needing, not only to pair the VS's in use, but also double them each time we add a new distinction. We may end up in the awful situation of needing:

> <not-equals, VS5> => glyph must be bold and angled
> <not-equals, VS6> => glyph must be non-bold and angled
> <not-equals, VS7> => glyph must be bold and not-angled
> ...

There is a further complication. Such a specification is pretty loose. Suppose a glyph has a slash with an angle of 32° from vertical. That clearly works for VS1 in the above case, and not for VS4. But how about 16°? 8°?, 4°?, 2°?, 1°? ½°?, ¼°? When does it cross over? Where do we draw the line? What happens when the character is italicized, obliqued? When we are specifying the behavior of a <character, VSn> pair, we have to provide enough information so that someone can tell whether they are following the specification or not, otherwise it is pretty useless.

An alternative approach is to say that the effect of VSn is targeted at distinctions *within* a single font, *not* across fonts. That is, if a font supports the sequence

> <not-equals, VS1> => glyph must have angled slash

what that means is that:

> The sequence <not-equals, VS1> must be visually distinguishable from not-equals alone *in that font,* and that the way in which it is visually distinguished from non-equals alone is in accordance with the description given in StandardizedVariants.html.

If we took this approach, then we would not be required to always add pairs (or worse yet, double the number of VS's) for each new distinction.

---

* Note: the term "glyph" can mean two things: a particular shape, or an abstraction over a set of shapes. The terms used to distinguish these two senses are *glyph image* vs. *abstract glyph*. I am using "glyph" in the former sense, whereby two different shapes are two different shapes. Thus a Palatino "b" is a different glyph than a Times Roman glyph. Some people mean the latter, but to my knowledge nobody anywhere has any criteria for establishing whether two particular glyph images belong to the same "abstract glyph" or not, so I consider it a completely useless concept.

Mark

_____

----- Original Message -----
From: <Peter_Constable@sil.org>
To: <unicore@unicode.org>
Sent: Tuesday, December 18, 2001 13:26
Subject: Re: A use for Variation Selector VS2

> I realise that this thread is ancient history (being a week old already),
> but I've just gotten around to reading it and thought I'd add my thoughts
> in case people are still thinking about it.
>
>
> Asmus wrote:
>
> >The remedy is obvious, and you mentioned it in your message: Don't give
> it a
> >blanket semantic, but use specific enumeration that includes a
> description of
> >the glyph (or glyph range).
>
> Clearly this is the right thing: we don't need a special semantic; we just
> need to refine the process for defining VS mappings. It seems to me that
> what we have done to now is (e.g.) assign X VS1 to mean "select the glyph
> with the slanted bar", having in mind an opposition to a glyph with a
> horizontal bar that happens to be what we used as the representative glyph
> in the chart, and which we are assuming as a default. But that is an
> invalid assumption about that representative glyph. **It is not a default
> shape from which the VS-mapped ones vary; it is only representative.** If
> someone needs to be able to specify "select the glyph with the horizontal
> bar", then that is a situation entirely analogous with X VS1, and we
> should define a mapping X VS2 for it (or X VS7 or X VS35 -- the number
> isn't important). In other words, whenever we are assigning VSs for glyph
> variants, we need to assign them for *all* the variants that need to be
> specifiable, including the one we were tacitly thinking of as the default.
> There is no default -- if we need to control the glyph, we need to assign
> it to a VS.
>
> =
> This avoids both problems that Ken mentioned:
>
> >My concern with this is twofold:
> >
> >1. It puts the glyph choice in the book into handcuffs, a constraint that
> > > we may come to rue later.
> >
> >2. It gives a *specific* semantic to one of the VS's -- something which
> > > we have otherwise refused to do. (E.g., we all agree -- I hope -- that
> > > assigning a glyph styling semantic to a particular VS would be a
> terrible

> > precedent.)
>
> No special semantic is used for any particular VS, and the representative
> glyph in the book continues to remain that: a purely representative glyph
> and not a specific glyph.
>
>
> - Peter
>
>
> ---
-
> Peter Constable
>
> Non-Roman Script Initiative, SIL International
> 7500 W. Camp Wisdom Rd., Dallas, TX 75236, USA
> Tel: +1 972 708 7485
> E-mail: <peter_constable@sil.org>
>