L2/04-213

Responses to Several Hebrew Related Items

Jony Rosenne, June 7, 2004.

Phoenician

While I do not intend to oppose the proposal, there is a problem that should be addressed: The Phoenician script (if it is a script - I don't want to go into that) had been also used to write Hebrew, and there is a certain volume of Hebrew texts in the Phoenician script. They were even sometimes mixed, as was the case that was mentioned with certain Dead Sea scrolls that used the older script for spelling out the divine name.

When used for Hebrew, the Phoenician script is equivalent to the Hebrew script, and in this case should not be distinguished for sorting and searching. The situation is not the same as Serbo-Croatian, where the same language is written in two different scripts, because in the case of Hebrew and Phoenician there is an extremely simple and precise rule - there is a one to one correspondence between the Phoenician characters and the corresponding subset of Hebrew, namely the 22 non-final letters.

Whatever is the decision of the UTC, this problem ought to be addressed.

Meteg

It should be noted that the issues discussed in the proposal are relevant to a specific use of Hebrew, namely Biblical Hebrew, and irrelevant to general use.

The UTC should consider any possible impact on other users of Hebrew. It should be specified that an implementation of Unicode Hebrew for general use is not required to handle CGJ, ZWJ, or ZWNJ, or to place the Meteg anywhere other than indicated by the combining class, or indeed to display it at all.

I would like to take this opportunity to bring to the attention of the UTC the following statement from Israeli Standard SI 4281, Information Technology: Implementation of Hebrew in the Hypertext Markup Language (HTML) (unofficial translation). While it does not contradict the letter of TUS 5.3, I suggest that a clarification of clause 5.3 is in order.

> **5.4 Rendering**
>
> This standard does not specify how to render characters that are impossible to display, such as Chinese or Japanese characters when there is no suitable font.
>
> **5.4.1 Rendering Hebrew Characters**
>
> Software applications shall render all the characters in SI 1311, and shall select one of the following alternatives to render the points and cantillation marks of SI 1311.1 and 1311.2:
>
> 1. Render them correctly. If the system can only render some of them only they shall be rendered. The correct rendering of the points and cantillation marks is a complex function of the base letter and the

collection of points and cantillation marks that join it and is out of the scope of this standard.

2. Do not render them. No indication should be given to the user that points and cantillation marks were not rendered, unless he explicitly requests it.

   Note: This is a special case of handling undisplayable characters according to HTML 4.

   This standard does not require the rendering of additional characters of the UCS, not even those characters that are associated in the UCS with the Hebrew script but are not included in Israeli standards.

   5.4.2 Handling Unrenderable Characters

   When a document is saved, the characters that are not supported shall be saved (including characters that are not rendered).

   When a document is transmitted, the characters that are not supported shall be preserved if the external encoding allows it. When it does not, they should be encoded as numeric character references [if the external encoding supports numeric character references].

   [Otherwise] If the external encoding supports only some of the points and cantillation marks, those characters that are supported shall be preserved and the others shall be suppressed with no indication of error.

The reasoning is that if a system that supports Hebrew does not support points or accents, it should present the text to the best of its ability, but there is no need to replace the points and accents it cannot render with square blocks or whatever is used to indicate a missing glyph unless the user specifically asks for it.

Holam

There are two different cases of the letter Vav with the point Holam. It could be a Holam Male, where the Vav is mute and together with the point represent the vowel, or it could be Vav Haluma - the letter Vav with a Holam Haser, where the Vav is the consonant and the point is the vowel.

In fine printing these two cases are distinguished. In the first case, Holam Male, the point is top center or top right relative to the Vav, in the second case top left. Actually, a Holam Haser on the Vav is identical in this respect to a Holam Haser on any other letter.

Both cases are normally encoded in Unicode as Vav Holam. In order to make the visual distinction, several people have adopted various stratagems, such that together with specific font designs the desired visual effect is achieved. Some put the Holam of the Holam Male before the Vav, some suggest the use of ZWNJ, CGJ or ZWJ in various combinations.

The result is an interchange incompatibility problem. This is a plain text issue, and should be addressed by the UTC.

The Holam Male - Vav Haluma distinction is theoretically sound. Holam Male should have been encoded Holam Vav rather than Vav Holam, because the base character for the Holam point is the consonant for which it is the vowel, not the silent Vav.

The Holam belongs to the consonant, and in fine typography is placed after the consonant and before the Vav, which could easily be viewed as if it were upper right of the Vav instead of upper left of the consonant. Look at the Lisbon Bible sample in Peter Kirk's paper. Compare the two Hebrew words that sound Bo, Bet Holam Alef and (according to this proposal) Bet Holam Vav. In the first, the Holam point, which clearly belongs to the Bet, is often placed over the right hand side of the silent Alef, and the same happens with the second and the silent Vav.

From a Unicode point of view, the base letter is the consonant, the combining mark follows it, and its placement relative to the Vav is a rendering/presentation/font issue - if the Vav has no other vowel it is Holam Male, if the Vav has its own vowel it is Holam Haser.

It should be noted that Holam Male is much more common than Vav Haluma.

The easiest and most correct solution, that requires no change to Unicode and no additional characters, is that those who wish to make the distinction just encode Holam Male as Holam Vav, assuming, of course, that in the fullness of time the fonts will be fixed. Although we are told it isn't easy, I believe that as rendering systems become more sophisticated it will be easier to do it. I still remember being told that automatic shape determination for Arabic was not feasible. The legacy data, using Vav Holam, will still display as desired and serve those who do not, will not or can not make the distinction.

Qamats Qatan

I came to believe that the UTC should consider the two questions of Qamats Qatan and Shva Na together.

The Hebrew vowel Shva has two meanings, known as Shva Na and Shva Nah. The Hebrew vowel Qamats has two meaning - Qamats Gadol and Qamats Qatan. Some printers desire to make the difference visible.

This is analogous for example to the dual meaning of the letter s in the English word summers, if a text book for teaching English would wish to make the distinction visible.

The two issues - the first has been discussed extensively and the last was only mentioned - have been around for some time. They all involve the same issue: certain users wish to make a certain distinction, while other users do not, and the common usage is not to make it. Also there is a legacy of data that does not make the distinction.

There is a difference between the Holam case and these two - the Holam distinction is theoretically sound.

Contrary to the discussions that I had seen (and I cannot claim to have read them all), in all these cases there are not two, but rather three, variants, because we must count also the undistinguished case which is common in general use.

We have been asked to add a character for Qamats Qatan, without a request for Qamats Gadol. But the existing Qamats serves for both. We have been mislead by the samples, that added a glyph for Qamats Qatan and used the existing Qamats glyph for Qamats Gadol (it is often a bad idea to judge by appearances). But we cannot and should not change the meaning of the existing Unicode Qamats - it serves for both. Therefore, if it is decided to accept Qamats Qatan as a new character there should also be with it a Qamats Gadol character.

The same goes for the Shva Na and Shva Nah.

An additional consideration against the proposal has come up in our discussions in Israel: There is no agreement which Qamats is which and which Shva is which. There are quite a few words with Qamats where some people think it is Qamats Gadol and some think it is Qamats Qatan, and the same with respect to Shva.

The consensus here seems to be that this is a glyph variant and not a different character.

Whatever the UTC decides, it should consider the possible impact on other users of Hebrew who do not wish or are not able to make these distinctions. These distinctions are not part of the general use subset of Hebrew. It should be clearly stated in relation to these additional characters, if they are accepted as such, that an implementation of Unicode Hebrew for general use, that implements the Hebrew subset, is not required to handle the additional characters or glyph variants of Qamats Gadol and Qamats Qatan, Shva Na and Shva Nah.

This leads us into another minefield: It requires a text transformation, where the Qamats Qatan and Qamats Gadol are both folded into a regular Qamats. This transformation is required when text that uses the distinguishing characters is transferred into text or a system that does not, for example by cut and paste, or by reading a file.

My recommendation is to decide that these are glyph variants, and encode them in some ignorable way.

Qere and Ketiv

The Hebrew Bible has for many words two versions, the "Writing" version (Ketiv) and the "Reading" version (Qere). The convention is to write the letters of the Ketiv version with the points and accents of the Qere version, possibly with a note in the margin.

When one tries to encode it as it appears there are many problems, because the number of letters may be different. In some cases there are fewer letters in the Qere version, so it appears that some of the letters of the Ketiv version have no points, in other cases the number of letters agrees but the points are ungrammatical. In more complicated cases there are more vowels and marks than there are letters, in a few cases there are no letters at all, just points and accents.

A common case is the name of the city Jerusalem, Yerushala(y)im, and its derivative Yerushala(y)ma, mostly spelled in the Bible without the second Yod, the Hiriq or Shva vowel of the missing Yod squeezed between the Lamed and the Mem, sometimes giving the illusion that the Lamed has two vowels.

Another common case is the divine name, where the Ketiv version is Yod He Vav He and the Qere version is either Alef Hataf-Patah Dalet Holam Nun Qamats Yod., or Alef Hataf-Segol Lamed Holam He Hiriq Final Mem.

This issue was discussed on the Hebrew and Unicode lists. I suggest the UTC should consider taking a stand. I don't think this issue is a plain text issue. Mark-up should be used to provide the two alternative texts. I don't believe it is possible or reasonable to computerize all the possibilities that are afforded the scribe when he manually places the points and marks of the Qere on a shorter Ketiv.