L2/17-149

Proposal to Encode
Arabic Hamza Above Isolated Form

Behnam Esfahbod – May 10, 2017

This document demonstrates the need to encode an isolated presentation form for `U+0654 ARABIC HAMZA ABOVE` character and proposes to encode this character in the `Arabic Presentation Forms-B` block of the Unicode Standard, next to other presentation forms for Arabic diacritics.

1. Background

In 1980s and 1990s, various visual (non-semantic and left-to-right) encodings existed for Perso-Arabic script. One of the most popular such encodings in Iran was “Iran System”. [IRAN_SYSTEM]

Zarnegar, a desktop publishing software mostly popular in the 1990s, with some versions still in use today, has used two different encodings for its source files. The first encoding used was an 2-form left-to-right (visual) encoding based on Iran System, which we call *Zarnegar1 character map*, lasting until the 1996-1997 version of the application (“Zarnegar 75”). Afterwards, Zarnegar switched to a 4-form bidirectional (semantic) encoding, which we call *Zarnegar75 character map*. [ZARNEGAR]

Despite the migration to a semantic encoding, there still exist plenty of Persian documents encoded in *Zarnegar1*. The new encoding, *Zarnegar75*, is still in use by the latest version of the software, still available in the market. Best practice for processing these documents is to convert the bytes to Unicode Arabic Presentation Form characters.

2. Missing Character

Arabic diacritic Hamza Above is used extensively in Persian text, mostly along with Arabic letter Heh. The semantic encoding for the letter is `U+0654 ARABIC HAMZA ABOVE`.

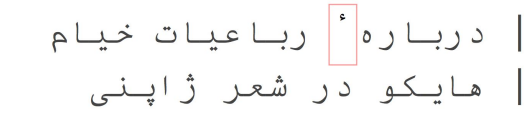

Figure 1: Arabic Hamza Above in its isolated form, as used in a Zarnegar document

Unlike other commonly used Arabic diacritic marks, `ARABIC HAMZA ABOVE` has no presentation form encoding in Unicode. *Zarnegar1* character map encodes Arabic Hamza Above in its "isolated form" in byte `0xB4`. (*Zarnegar75* character map also encodes this character, but exact encoding of it is still unknow.)

This presentation form is similar to `U+FE70 ARABIC FATHATAN ISOLATED FORM` and the rest of the `ISOLATED FORM` diacritics (`U+FE72`, `U+FE74`, `U+FE76`, `U+FE78`, `U+FE7A`, `U+FE7C`, and `U+FE7E`), where the diacritic is presented in its own space, with a `Letter, Other [Lo]` General Category property. (See Figure 2.)

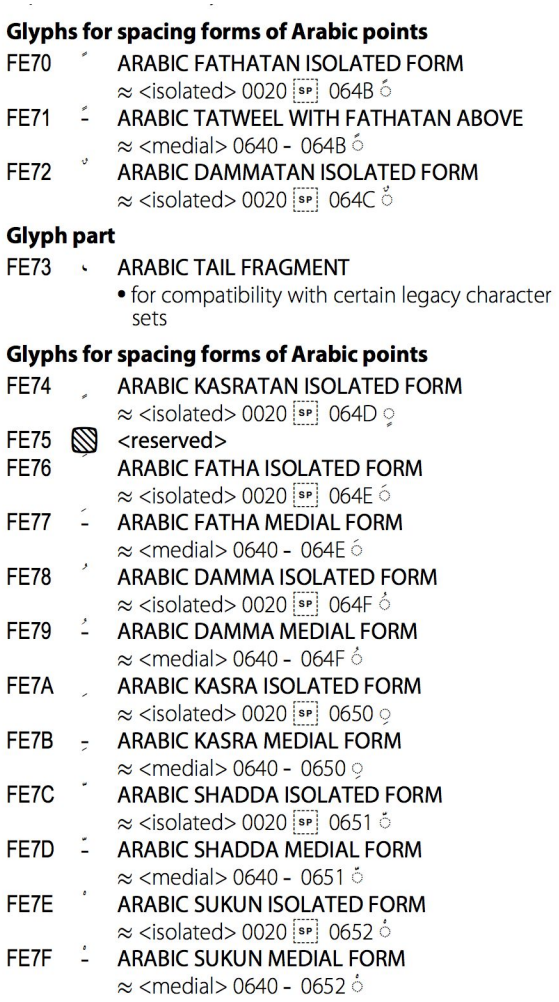

Figure 2: Pairs of isolated and medial presentation forms of common Arabic diacritics, except for Arabic Kasratan, which only has isolated form.

Although this character looks similar (and would be decomposable to) a `<U+0020 SPACE, U+0654 ARABIC HAMZA ABOVE>` sequence, the sequence is not suitable for processing because of the information that gets lost. When converting this visual encoding to semantic encoding, the actual mapping depends on the position of the

character, as it happened in Zarnegar rendering. If ARABIC HAMZA ABOVE ISOLATED FORM appears after an Arabic letter, it should be converted to <U+0654 ARABIC HAMZA ABOVE> and act as a mark on the letter, otherwise, it should be converted to the <U+0020 SPACE, U+0654 ARABIC HAMZA ABOVE> sequence or something similar.
For example, as shown in Figure 1, when ARABIC HAMZA ABOVE ISOLATED FORM appears after letter Heh, it has the implicit meaning that the Hamza is placed above the letter.

3. Proposal

The proposed solution to the problem is to encode an isolated form for Arabic Hamza Above with the following properties:

- ;ARABIC HAMZA ABOVE ISOLATED FORM;Lo;0;AL;<isolated> 0020 0654;;;;N;;;;;

3.1. Code Point

There are three unassigned code points in the Arabic Presentation Forms-B block:

- U+FE75: appears to be reserved for a medial form of Arabic Kasratan.
- U+FEFD: proposed to be used for ARABIC HAMZA ABOVE ISOLATED FORM.
- U+FEFE: proposed to be reserved for medial form of Arabic Hamza Above (ARABIC HAMZA ABOVE MEDIAL FORM) if and when needed, similar to U+FE75.

4. Encoding and Sample Data

More details of *Zarnegar1* character map and sample data file are available at https://github.com/behnam/python-zarnegar-converter

A. Acknowledgements

Thanks to Cecil H. Green Library of Stanford University, specially John A Eilts and Behzad Allahyar, for sharing their collection of Zarnegar documents.

B. References

[IRAN_SYSTEM]
https://en.wikipedia.org/wiki/Iran_System_encoding
[ZARNEGAR]
https://en.wikipedia.org/wiki/Zarnegar_(word_processor)