Title: Script property of characters with multiple scripts

Author: Roozbeh Pournader and Behdad Esfahbod (Google)

Date: 2012-11-04

Action: For UTC's action

Suggested action

For characters that have a Script_Extentions property that contains **more than one script**, change their Script property from Common or Inherited to one of those scripts if that script is predominantly and uncontroversially what constitutes the **overwhelming majority** of usage of that character.

And keep this as a guideline going forward with both old characters that will be having a Script_Extensions property and new characters.

Alternative action

Come up with a clear alternative guideline, document that, and keep it consistent going forward. For example, we could also introduce some further structure into Script_Extensions property values, adding an additional optional primary script for some characters.

Background

At the moment, characters that are shared across various scripts have their Script property inconsistently defined. For example, both U+0660 ARABIC-INDIC DIGIT ZERO and U+FDF2 ARABIC LIGATURE ALLAH ISOLATED FORM have their Script_Extensions property set to Arabic and Thaana, while the digit has its Script property set to Common and the ligature has it set to Arabic. For users of these characters in both Arabic and Thanaa scripts, the difference is arbitrary. For all practical purposes, the ligature is not more Arabic than the digit, nor is the digit more Thaana than the ligature. They are both imports from the Arabic script used in Thaana.

At the same time, it is very useful to be able to arrive at a default script for some common characters if the characters are seen in an out of context situation. For example, the sequence <U+0640, U+064B> is commonly used to refer to a Shadda, properly rendered as:

But an implementation trying to render that sequence is left to its own devices to figure out what script (and font, and font tables) it should use to render the sequence if no other information is made available. At the moment, various implementations of the Unicode Standard, when they encounter such a sequence out of context, render it as:

This is an artifact of both Tatweel and the Shadda having their script property set to Common or Inherited, resulting in the rendering system not knowing that it should actually look at Arabic-related font tables in order to apply positioning information (for example as defined in OpenType fonts and their GPOS tables).

We have considered hard-coding overrides to make characters fall into certain scripts, but we believe such a solution doesn't scale for the various minority and historical scripts that we plan to support. Most such user communities should be able to get their writing system display properly with standard properties properly defined in the Unicode Standard and proper fonts, without needing to wait for text layout engines forever to special case a few character for them.

We also considered using block information, but we believe that's just a hack too: characters like U+FEFF (BOM) happen to be in an Arabic block, with no relation to Arabic. We would still need to special-case characters.

Having the Script property set to a value that is usable has the benefit that implementations could use it as a first fallback when there are no other signals hinting towards one script.

But even if UTC does not agree with our main proposal, we would appreciate it if it could come up with a more consistent approach, or clarify and document when certain common character get a specific script and when not, and conform to it.