L2/17-019

Re: UTS#46 Issues

From: *M. Davis*

Date: *2017-01-17*

Draft: *https://goo.gl/Kqvpfh*

We need to consider issues raised in http://www.unicode.org/review/pri317/ and related email so that we can make any necessary changes in version 10 of UTS#46 (June 2016). The following is a proposal for the January UTC meeting, so that it can go out for public review at the same time as the Unicode 10.0 beta. (While #46 is not part of Unicode 10.0, it is sync'ed in terms of schedule and version.)

Each entry below summarizes the original issue in italics, provides some background, then provides a proposal to resolve the problem.

Contents

A. [Valentin Gosu]

"We recently implemented UTS46 support in rust-url which is used in servo. However, it seems that other browsers don't enforce validation rule 2, regarding - in the 3rd and 4th positions of the label. There are actually some URL which match this in use on youtube.com More info in: https://github.com/whatwg/url/issues/53."

The validation rule was copied from IDNA2003/2008... As I recall, the purpose was for the IETF to have the latitude to introduce additional prefixes should they need to later; I think there was some discussion of different kinds of IDNA or versioning that would motivate that. It is possible that the IETF may want to make some use of it in the future, although it seems quite unlikely that it would be used for the original imaginings.

Later feedback showed that all of the browsers also allow leading and trailing hyphens, so probably both rule 2 and rule 3 can be covered by one flag.

Proposal:

1. *Add an input parameter to "**Main Processing Steps**" (unicode.org/reports/tr46/#Processing) and callers: AllowHyphens. If true, it causes steps 2 and 3 of unicode.org/reports/tr46/#Validity_Criteria to be skipped. (Alternatively, we could call it CheckHyphens, and change the polarity; 2 & 3 are only checked if it is true.)*

B. [Marc Lehmann] [Alastair Houghton]

(Errors in IdnaTest.txt)

Proposals:

- *It appears that Marc's issues were addressed, but we should contact him to verify.*
- *Fix problems exposed by Alastair.*

C. [jcranmer]

"Oh yeah, I came back into this and recall that the IdnaTest.txt is really bad at telling you how to process it .... The ToASCII column uses nontransitional processing (read IdnaTest.txt's commented header) and

UseSTD3ASCIIRules=true (see §8 of the input). However, they definitely appear to have some extra rules not described in their algorithm (for example, ToUnicode should never produce an [A4_1] or [A4_2] error, since those are specific to the ToASCII regime and ToUnicode never calls ToASCII, yet you can clearly see for yourself that they do).”

Proposal:

- *Fix the description of IdnaTest to make it clearer exactly how it expects the test to be processed. (And make sure that there are no “extra rules”)*

D. [SimonSapin][annevk]

“If the URL Standard is to define interoperable algorithms, I think it needs to define in which requirements Section 4.1.2 sets the error flag.”
“It turns out that some of the implemented processing steps are SHOULD-level requirements in UTS46. I think these should be refactored to be options for the ToASCII and ToUnicode algorithms, so the URL Standard can enforce them. It does not seem great to allow variable processing.”

I agree with these suggestions, and for the reasons given.

Proposal:

- Wherever UTS #46 has a “should”, add an extra input parameter to control that option in the relevant processing. At this point, it appears that the ones to add are Bidi, ContextJ. So we could add two parameters:
    - CheckBidi
    - CheckJoiners

Question on ContextJ

This brings up two related issues:

1. The ContextJ rules for ZWNJ and ZWJ are pretty blunt hammers. Should we instead point at the rules in http://www.unicode.org/reports/tr31/#Layout_and_Format_Control_Characters which are more narrowly defined?
2. Also, given that some programming languages allow emoji identifiers, and emoji may contain ZWJs, should we update tr31 to allow them? Note that if the identifiers’ allowed characters already disallow emoji (such as IDNA2008), then that update would have no effect on them.