L2/22-088

Status report of the source code working group for UTC #171

To: UTC
From: Robin Leroy, Mark Davis, Source code ad hoc working group
Date: 2022-04-10

The source code ad hoc working group was created by consensus 170-C2 of the UTC, on the recommendation of the Properties & algorithms group, as described in document L2/22-007R2, section “Proposed Plan”, with Mark Davis as the chair.

I. Proposals

While there is much work remaining to be done, the working group recommends the changes described in the documents below for Unicode 15.0. These are relatively small changes in UAX #9, UAX#31, UTS #39, plus a change in the data files for UTS #39, for Unicode 15.0.

1. L2/22-087 Profile Changes in UAX #31 / UTS #39
    - This document proposes changes to address issues raised in the discussion of L2/22-082 (*Proposal for an option in UAX #31 to prohibit ZWJ/ZWNJ for identifier security*, Asmus Freytag and Michel Suignard).
2. L2/22-072R Proposal for amendments to UAX#9 and UAX#31
    - This document proposes some non-normative changes to the documents that provide more background and examples where bidirectional identifiers (not just bidirectional controls) can cause problems in software.

II. Progress

The working group reports on its progress along the five goals listed in the section “Proposed Plan” of L2/22-007R2:

A. Engage with MITRE to get more accurate wording into the CVE records.

Proposed wording has been drafted by working group member (and UTC chair) Peter Constable. Mark Davis and Peter Constable have had initial discussion with contacts at MITRE and this item is in progress.

B. Assemble documentation providing guidance for avoiding spoofing issues. Make that available for review and feedback.

The working group has been collecting some principles that should govern the desired behaviour of tools that implement mitigations for issues of misleading source text display. These will serve to constrain the space of effective and acceptable mitigations.

For instance, on the subject of displayed order, there are lexical units, which we call "atoms" (they may be smaller than language's tokens, *e.g.*, they include string or comment delimiters) for which the following should hold:

1. They should not be split in rendering;
2. They should be ordered in some order (LTR or RTL), fixed throughout the document, that matches the memory order;
3. Their display might depend on their nature (the contents of a string may render differently from the same text as a numeric literal), but not on the surrounding atoms.

These are more precise elaborations of HL4 in UAX #9 Unicode Bidirectional Algorithm. As we point out in L2/22-007R2, these principles are followed by Visual Studio; they have recently been applied to Visual Studio Code as well.

On matters of confusability, we found that it is a problem if visually equivalent identifiers are logically distinct, but also if logically equivalent identifiers are visually distinct (at least in case-sensitive languages); this is relevant for instance to the proper treatment of ZW(N)J, and to the choice of Normalization Form; see below.

C. Produce Unicode documentation, such as draft proposed updates of UAX #9 ("Bidi", aka UBA), UAX #31 ("Identifiers"), UTR #36 ("Security"), and UTS #39 ("Security Mechanisms") using the information in B, and post for comment.

There are some initial changes proposed by the working group, in the Proposals section above.

Once it becomes clearer what the necessary recommendations are, the working group is considering producing a dedicated document consolidating recommendations for handling source text. This could be a UTR or UTS, depending on whether normative text is needed.

D. In ICU, respond to tickets filed, and provide code snippets and/or APIs to implement utility functions that could be used directly to help avoid problems. (The implementations could also be ported to other languages.)

Working group member (and ICU-TC chair) Markus Scherer has responded to ticket ICU-21830. Providing code snippets and/or APIs is contingent on concrete recommendations (C.).

E. Examine whether new properties and/or property values, or changes to values, would be useful.

The working group has no concrete propositions for new properties at this time; defining a stricter kind of confusable (one which would, e.g., treat **γ** (*gamma*) and **y** as distinct, but **у** (*cyrillic u*) and **y** as confusable) has been discussed and may be recommended at a later time.

A change to Identifier_Type and Identifier_Status values for ZWJ and ZWNJ is proposed for Unicode 15.0, but other changes may be proposed in the future, as part of broader recommendations from the working group. The group is looking at the interplay between those characters allowed in identifiers, and those characters whose usage in identifiers could be disallowed or warned about. The treatment of identifiers may also depend upon the context. For example, an identifier declaration might be allowed with no warning, except when it is confusable with another identifier in the same scope.

III. Other Documents

1. Document L2/22-028 (*Bidi in programming languages and markup languages*, Kent Karlsson) was brought to the attention of the working group. The working group noted that some of the ideas explored in that document were already under consideration, and recommends that the UTC take no action.