L2/22-161

Status report of the source code working group for UTC #172

To: UTC
From: Robin Leroy, Source code ad hoc working group
Date: 2022-07-04

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 we plan to bring some more extensive proposals to UTC #173, our current focus is to address remaining issues for Unicode 15.0. Indeed, while incorporating the changes proposed by this group at UTC #171 (see L2/22-088) into the standard annexes and technical specifications, we found some inconsistencies in UAX #31, which have been called out by review notes in Unicode 15.0β.

I.1. Addressing inconsistencies in UAX #31

Document L2/22-110 "Addressing inconsistencies in UAX #31" is a proposal to correct these inconsistencies, for Unicode 15.0.

I.2. Numbering paragraph requirements in UAX #31

The group further recommends that, in UAX #31, in requirements comprising multiple paragraphs, these paragraphs be numbered instead of subsequent paragraphs being marked by bullets (as is currently the case in R1, R1a, R8) or unmarked (R2, R3). The following illustrates the proposed change for UAX31-R1.

> ***UAX31-R1**. **Default Identifiers:***
>
> *(R1.1) To meet this requirement, to determine whether a string is an identifier an implementation shall use definition UAX31-D1, setting Start and Continue to the properties XID_Start and XID_Continue, respectively, and leaving Medial empty.*
>
> *(R1.2) ~~-~~Alternatively, it shall declare that it uses a **profile** and define that profile with a precise specification of the characters that are added to or removed from Start, Continue, and Medial and/or provide a list of additional constraints on identifiers.*

II. Progress

The working group reports on its progress along the five goals listed in the section "Proposed Plan" of L2/22-007R2. Items already addressed at the time of UTC #171 (see L2/22-088, section "Progress") are struck through.

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

Mark Davis and Peter Constable have had discussions with contacts at MITRE; both CVE-2021-42574 and CVE-2021-42694 have been updated to incorporate both the original wording and the Consortium's suggestion, with a ** DISPUTED ** mark (as per MITRE policy).

This goal is complete.

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

The plan of the group is to produce a new Unicode Technical Specification focusing on source text handling, a draft of which would be presented to UTC #173.

This document would comprise the following:

I. Recommendations for programming language syntax.

   This section would focus on the application of UAX #31 (Identifiers and Pattern Syntax) to programming languages specifically. That UAX has a broader scope; *e.g.*, it includes hashtags, and it can be difficult for programming language designers to navigate the interplay between the different requirements, especially in the presence of profiles (profiles being customizations of the default syntax). Standard profiles would be provided for common use cases. They would either be defined in the new UTS or in UAX #31.

   An annex would deal with questions of migration, such as introducing normalization in a new version of a programming language, changing normalization form, or switching from UAX31-R2 "immutable identifiers" to UAX31-R1 "default identifiers". Indeed, the group has found that many major programming languages are considering such changes, or would benefit from them; but the implications are complex, as care must be taken not to change the meaning of a program between successive versions of the language.

II. Recommendations about source text display.

   This would include recommendations about bidirectional ordering, such as the application of higher-level protocol HL4 of UAX #9, but also recommendations about the display of invisible and blank characters: the circumstances in which they should or should not be made visible, and the way in which they should be made visible if they are.

III. Recommendations for tooling and diagnostics.

   This would include recommendations and algorithms for more general diagnostics, such as confusable detection. It would also include recommendations and algorithms for tools that "fix" the code, e.g., by inserting implicit directional marks between tokens.

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.

Proposed updates will be presented to UTC #173.

These documents would be updated to support and refer to the new UTS; it is likely that the algorithms and data supporting confusable detection in UTS #39 would be updated, and that some material would move from UAX #31 to UTS #39 so that the separation of concerns between the two documents is clearer. The

changes to UAX #31 would likely also include “standard profiles” (changes to the default identifier syntax) for reference in the new UTS.

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.)

The plan of the group is to provide implementations of the algorithms described by the new UTS to ICU, and updated implementations for existing algorithms.

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

The details of specific property changes will become clearer once we produce our drafts.

One specific change being considered by the working group is to add the ZWJ and ZWNJ characters to the set [:Other_ID_Continue:], which would bring them into [:XID_Continue:], that is, into default identifiers absent security considerations.

Background

We have determined that this is unlikely to have a security impact, as the equally invisible variation selectors are already part of [:XID_Continue:]; they must remain in XID_Continue by the stability policy:

> *Once a character is XID_Continue, it must continue to be so in all future versions.*
>
> List of default ignorable code points in XID_Continue;
> List of default ignorable code points not in XID_Continue.

Users of the General Security Profile would disallow both the joiner controls and the variation selectors following the changes in L2/22-087 *Profile Changes in UAX #31 / UTS #39*.

At the same time, these characters are more orthographically useful than the already-allowed variation selectors, and potential usability issues arising from their presence in source code would be adequately mitigated by our new recommendations. Further, the distinct treatment of the conceptually similar variation selectors and joiner controls in UAX #31 has led to confusion even among experts, so this would improve the readability of the standard.