ISO/IEC JTC1/SC2/WG2 **N 2169**

Author: Kent Karlsson
Status: Expert Contribution
Date: 2000-03-02

Comments on 'Language Tag' characters for 10646-2, plane 14.

These are some a bit more detailed comments supplementing the Swedish NB NO vote to the CD1 ballot on 10646-2. (See also N2168 on 'Math Alphanumeric' characters.)

The "language tag" characters (a kind of control characters) are suggested to be allocated in the "special purpose plane" (plane 14).

The Swedish comment is to remove all text and tables referring to plane 14; including annex D. The presence of the 'language tag' characters (in plane 14) is one of the reasons for the Swedish NO vote on the 10646-2 CD.

This document gives motivations for the NO vote, motivations that for brevity are not included in the vote itself.

a. History

These "language tag" "characters" were once construed in order to fight down a technically problematic proposal that used (otherwise) ill-formed UTF-8 sequences as language tags. Claims have been made that this (or, later, the "tag" characters) would be needed for Internet protocols. This is not true, see point c below. Language tagging is, and remains, a 'higher level protocol' issue, not an issue deserving new characters (with syntax) dedicated to language tagging.

b. Acknowledgement of the need for language tagging

The very strong objection, via the Swedish NO vote, to the suggested "language tag characters" does in no way stem from a view that language tagging never should be done. Language tagging has its place, and should be properly supported. But NOT via the "language tag characters", as explained below.

c. Internet protocols

Higher level Internet protocols often consist of a "conversation" in plain text supplemented with data (which may be text or other data). The language tagging needs for Internet protocols concern the "conversation" part, in particular things like error messages. For the other data, if it is text, e.g. XML files, other kinds of language tagging are often already in place, or could be put into place (like when the response is a list of names or addresses). Some Internet protocols already provide language tagging for the "conversation" part, but do so using ordinary characters (for letters mostly, plus HYPHEN-MINUS). See for instance RFC 2596, *Use of Language Codes in LDAP* (ftp://ftp.isi.edu/in-notes/rfc2596.txt). So for these Internet protocols no new "tag characters" are needed. Other Internet protocols are not directly generalisable to use language tagging with ordinary characters. It is for these that the “tag characters” are purportedly needed. However, incompatible (in the sense that there are new features that MAY be used) protocol upgrades have been done before: most notably the SMTP protocol upgrade to ESMTP. Or the replacement of POP with IMAP (using a different replacement strategy). There is no reason to believe that such protocol replacements cannot be done again. Indeed, such replacements need to be done anyway to take advantage of any new “tag characters”, which effectively removes the reason for allocating any “tag characters” in the first place.

d. Optional alternative suggestion: a META combining character

If some kind of “meta-characters” based on graphic characters is to be introduced, that should be done in a general manner. It suffices to encode a single character to achieve this, and there is no limit on which graphical characters that may be turned into meta-characters this way: encode a single new character, META, or, if you prefer, call it COMBINING META, which is a combining character that should occur immediately after the base character in a combining sequence. It makes the entire combining sequence into a meta-character. META-generated meta-characters can then be used to spell out whatever in-line graphic characters based higher-level text ‘protocol’ desired, including language tagging. Moreover, adding a single META character is much more general than the current proposal for tag characters.

e. Search/match

One claimed advantage with having special characters for language tags is that it would simplify search (match) operations. It would be easier to avoid false (preliminary) matches, which had to be further examined and then possibly rejected. That is not quite true, since even if the data is in (Unicode) normalisation form C, which uses mainly precomposed characters, there may still be some combining character after the initial match that needs to be examined before one can determine if the match can be accepted. Therefore, for META characters (as suggested in point d above), their method of detection would have to be there anyway.

f. Suggested future development of Internet protocols

Follow the lead of RFC 2596, *Use of Language Codes in LDAP*, and use language tags expressed with already allocated graphic characters. In case that is not immediately feasible as an extension of current syntax, aim for replacing the protocol with a new one where such language tags fit in. Alternatively, and if allocated, a META character (see point d above) could be useful.

g. Higher level mark-up, e.g. XML

These usually have, and XML has, its own language tagging mechanism, in the case of XML using characters for letters plus HYPHEN-MINUS. Use of the proposed “tag characters” must be forbidden in this context, or at least the “tag” control characters must be ignored. Thus, the suggested “tag characters” do no good together with higher-level mark-up, and may interact badly with other language tagging if the tag characters are interpreted. Indeed, they constitute is an unnecessary complication, which we are better off without.

h. Plaint text language tagging

Some suggest that the “language tags characters” are intended for plain text (which was not the original intent, see point a above). However, using them with plain text constitutes a misuse, since language tagging is not “plain”, it constitutes mark-up. Italic, bold, and size changes are much more “plain” than language tagging.

j. Conclusion

Language tagging is not a plain text issue, and should not become one. Language tagging as such has it’s place and is useful, but the plane 14 tag characters approach is highly inappropriate. If absolutely necessary, allocate a general combining “META” character, to be used in any high level protocol where one likes, including but not restricted to such for language tagging. But the META character should be general, and any syntax for using it is out of scope for 10646. To make it easier to handle, a combining META character should come immediately after its base character (if not initially so, this can be achieved by the Unicode combining sequence normalisation reordering).

-----------------------------------end of N2169--------------------------------