Subject: A security concern on the planned modification to IDS definition
Source: Masahiro Sekiguchi (expert's individual contribution)
Date: 2012-2-17
Re: WG 2 N4234 (Proposed Changes to ISO/IEC 10646 Annex I Ideographic description characters)

Summary

WG 2 N4234 proposed removal of the length restriction from the IDS definition, and WG 2 seems agreed on the point. However, I have a concern on the decision.

I believe we should keep *some* restriction on the length. Instead of allowing any unlimited long IDSs, I propose to set a longer but reasonably small limit, e.g., 64.

The history

The original IDC/IDS proposal, WG 2 N1782 dated 1997 (http://std.dkuug.dk/JTC1/SC2/WG2/docs/n1782.doc), included no limit of the IDS length. It is WG 2 who set the restriction based on the experts' inputs. I was not there or I couldn't find any written document discussing the point, but I personally remember Takayuki Sato, who was a member of Japanese delegation then, told me what was discussed during the meeting.

As far as I remember, the reason WG 2 set the restriction was to allow small systems with limited resource, e.g., embedded micro controllers, to handle UCS data including IDS. Note that the nature of recursive nesting structure of IDC/IDS requires some working storage proportional to the length of IDS to be allocated when validating it (i.e., Are all internal IDSs nest properly? Are all IDCs followed by correct numbers of DCs?) By restricting IDSs to a small length, implementation can easily allocate a fixed small sized working storage.

A security concern

Today's embedded systems enjoy far more resources than those in 1998, so the pressure to keep the required storage small may be loosen. I believe, however, we have another requirement today: a security.

If we allow arbitrary long IDSs in our standard and a program tries to validate them fully, the program needs to prepare arbitrary large storage because the required storage is proportional. Practical implementation should set its own limit and make sure the input doesn't exceed the limit by its own way. The industry learned in the last decade or so that such type of storage management or sanitization is very often implemented badly, causing buffer overrun or other serious security halls.

The current limit of 16 characters is sufficiently small and I believe the implementation needs no complex management. Setting longer but reasonably small limit will satisfy the requirements in N4234, keeping the simple structure of existing implementations.

A proposal

Instead of removing the length limit, update the limit to a larger value. The new limit should be sufficiently large to cover known longest examples but should be kept reasonably small.

I propose a new limit of 64 characters, because it seems sufficiently long to write IDSs and reasonably small to implement.