**NCITS-L2/98-042R**
**ISO/IEC JTC 1/SC 2/WG 2 N 1724**
**Date: 1998-03-05**

**ISO/IEC JTC 1/SC 2/WG 2**
**Universal Multiple-Octet Coded Character Set (UCS)**

**Title:** **Formal Criteria on Disunification**
**Source:** **The Unicode Consortium, US National Body and Asmus Freytag**
**Action:** **For consideration and acceptance by SC2/WG2**
**Distribution:** **NCITS-L2, UTC, ISO/IEC JTC1/SC2/WG2**

*Text version 003*

*(Retyped from Hard Copy and an older version of this document – Uma).*

This document proposes a formal criteria for the evaluation of certain kinds of character encoding proposals. It is not intended for use with Han characters.

## Background

There have been repeated proposals to disunify existing characters. These proposals cannot be fully evaluated without a more rigorous framework concerning the disunifcation / unification of characters. Without such formal criteria, all decisions are 'ad-hoc' and different proposals may get different level of review. Bot the Unicode Technical Committee and ISO/IEC JTC 1/SC 2/WG 2 need to spend some time in evaluating and possibly formalising the criteria that we use to decide these cases. This is similar to the formalization we have done for script prioritisation, but uses different criteria.

NOTE: The unification criteria used for the Han script are very thorough and quite sufficient. This document attempts to establish formal criteria for use in other scripts. There is no attempt to change the procedures used in Han unification.

## What is disunification?

Disunification is the introduction of a new character which can also be encoded by an existing character. A strong case of disunification occurs where there is prevalent practice of using the existing character. A weak case of disunification occurs where there is little or no use of the existing character for the purpose for which the new character is intended.

> *Example*: Adding a period in a new script is a weak disunification if we assume that nobody has an existing implementation of that script using the regular period. Adding a clone of a Latin letter for use with Cyrillic script is a strong disunification as mixed Latin/Cyrillic character sets exist and have almost certainly been used for encoding the languages that the new characters are intended for.

## Cost and Benefits

Proposals always claim that disunification brings benefits. Formal criteria attempt to critically evaluate those benefits, but also compare them to the costs. Any disunification, especially strong disunification, introduces several types of cost to *all* complete implementations of the Standard.

- First, any complete implementation will have to add and support both an additional entry in the properties as well as an additional glyph, or glyph mapping for the disunified character.
- Second, whenever the character in question has no appearance distinction, there is the cost of accidental confusion and mis-identification. All implementations will need sophisticated handling of

equivalencies, especially, where disunification occurs on well-established characters (as opposed to among the characters of an entirely new script being fine-tuned in the proposal stage).
- Third, keyboards that support the disunification need to be widely (and by default) available, this is especially troublesome for strong disunification of Latin characters as most keyboards have a Latin layer from which it is easy to type the existing and now-disunified character.

## Criteria of analysis

### I. Costs

The following questions are designed to evaluate the costs associated with the disunification.

1. Is there a glyphic distinction?
2. Is there a behaviour difference?
3. Is the use of the new character restricted to a new context (for example, use with a novel script)?
4. Is the use of the existing, ambiguous character instead of the proposed new character common, prevalent or established practice?
5. Does the character exist in ASCII (ISO 646IRV)?

### II. Benefits

- First, appearance: does disunification help to allow multilingual monofont text in an environment where this is commonly needed? In what way?
- Second, layout: does disunification solve common layout differences (this would mostly be true for punctuation)?
- Third, searching/sorting: Is there a *common* case where disunification allows better support for these?
- Fourth, mapping to another standard: Is there a widely used standard that disunifies the characters in question? Are the characters in question the *only* ones that prevent cross mapping?

### III. Alternatives

Finally, the analysis must explore whether other alternatives are possible.

Can the desired effect be achieved by changes to the display layer?
Can the desired effect be achieved by changes to protocols?
Can the desired effect be achieved by processing algorithms?

### Examples of Precedents

**Character**: *Generic Decimal Separator Mark*

In 1991 the proposal was made to add a new punctuation character in the General Punctuation block that would have the semantic property of decimal separator, but could be imaged as either period, comma, space or apostrophe depending on the locale.

*Asserted benefit:* Solve the locale dependent display of numbers.

*Costs:* This new character would have disunified four widely used characters. Mapping from existing character sets would have become locale dependent. Users would have to turn on a special show-invisible-character mode to distinguish the new character from existing characters. Such modes exist, but are limited to word processing software, where numbers usually occur embedded in text, which in turn is 'frozen' into a given language. Database software, where locale dependent numeric displays are much more of an issue, does not normally need or support a visi mode. Finally, in 1991 there were no keyboards supporting this new character, but it would be needed in *all* languages and applications, and *all* software would have to be specially adapted for it.

*Alternatives:* There already is an established technology to deal with locale differences, and in a way that is not limited to decimal numbers.

*Result:* **Rejected.** The costs far outweigh the benefits.

**Character:** *Angstrom Symbol*

*Asserted benefit:* Provide roundtrip mapping for East Asian character sets.

*Costs:* This character disunifies A WITH RING, which is in wide use in only a limited number of languages that all use Latin-1. In the Latin-1 context, it would be natural to use A WITH RING as the Angstrom Symbol. The Angstrom unit is not one of the preferred powers for the metric units of SI, but it is still commonly used in some disciplines as it is convenient for atomic length scales. Disunifying the A WITH RING adds the important round trip mapping capabilities for East Asian character sets, but makes it harder to use the Standard as a pivot between these character sets and Latin-1. However, almost none of the other SI units that have explicit character codes in East Asian character sets can be mapped 1:1 with Latin-1, so the Angstrom Symbol adds little to that problem. Searching needs to support equivalencies, however, in the East Asian context the need for extended equivalencies (beyond simple case equivalence) is common.

*Alternatives:* None.

*Result:* **Accepted.** The benefits far outweigh the costs.