

Requirements for adding and updating Unicode properties

The problem:

Character properties are expensive to maintain. As properties multiply, so do the efforts to keep the UCD up to date. When characters are added, it is not always clear what values certain properties should take—as a result, many times characters are initially assigned an incorrect property value.

When property values are changed to address discovery of such issues at a later date it incurs costs due to instability. In some cases requested changes appear ‘logical’ but in fact violate the original assumptions for the definition of the property, in effect substituting a similar but different property.

The proposed solution:

- 1) Require that a definition exists in writing, and is agreed upon by the UTC, before adding a new property to the UCD. The definition needs to be specific enough to
 - a. help in assigning the property to new character codes by
 - i. providing a ‘domain’, i.e. define the types of characters that need to be reviewed for specific property assignments
 - ii. providing a ‘default’, i.e. the value the property should take for an ordinary assigned character
 - iii. providing a basis for deciding what property value to assign to any character
 - b. support the ability to review existing assignments
- 2) Require that new properties be considered provisional for one or more versions, to make it easier to change values while experience with using the property accumulates—later, make it harder to change the property, in order to provide as much stability for implementations as possible.
- 3) Require that proposed corrections be reviewed against the definition to ensure a continuity of purpose and to avoid ‘definition drift’.
- 4) Complete the process for existing properties lacking detailed enough definitions.