# Review of Identifier_Type for existing characters

***Review document for PRI #517 --- last updated: 2025-02-04***

The Unicode Consortium is broadly reviewing the assignment of Identifier_Type to characters of non-ideographic scripts listed as **Recommended** in UAX31. As published in the datafiles for UTS39, synchronized with Unicode Version 16.0, there are over a thousand non-Han characters with Identifier_Type=Recommended, for which feedback received suggests that the existing classification may be inappropriate for the purpose of defining a recommended repertoire for default identifiers.

Many of these characters might be better classified as **Technical** (not for general use in the writing system), **Obsolete** (no longer in current use) or **Uncommon_Use** (not attested as needed for an orthography that is in widespread everyday common use for any living language).

The Unicode Consortium is interested in obtaining additional information that would help improve the assignment of Identifier_Type. In particular, if any of the characters listed in documents L2/25-033 and L2/25-034 can be attested as being in use for an orthography that is in widespread everyday common use, we would like to receive feedback to that effect, preferably with a citation that documents such usage.

Likewise, clear indication that a character is either no longer in current use (Obsolete) or typically only used in a specialized context (Technical) would be appreciated. Please submit your feedback by the closing date of the PRI.

The feedback should prioritize the characters listed in documents L2/25-033 and L2/25-034, but in principle all assignments of Identifier_Type are open to reassessment if new and better information becomes available. The identifier type adjustments proposed in these documents at this point just constitute a body of feedback and do not predict the final outcome.

After a review of all available information, the Unicode Technical Committee will publish an updated data file with the updated assignments for Identifier_Type, which will then be subject to the normal beta review.

**Notes**

Identifier_Type values are used to guide implementers in setting rules for secure but usable identifiers. In particular, the recommendation is to not include characters that are unfamiliar to users because they are Technical, Obsolete or not in common use. There is a tendency for users to misinterpret an unfamiliar character as an unusual rendering of some more familiar character instead.

A typical example for that is the Old English letter Wynn 'ƿ' that most modern users of the English language would treat as a 'p', because the exact details of the bowl of a 'p' are not normally important.

The effect of identifying additional characters as Technical, Obsolete or Uncommon_Use is to remove them from the set of characters that are recommended for default identifiers. There is no stability policy that prevents such changes. If any system supports identifiers that have been created or registered before such a change and that are no longer considered default identifier, the suggested treatment would be to grandfather existing ones, but not allow the creation or registrations of new ones.

Implementations are also free to explicitly deviate from the recommended default identifiers as needed to serve a particular constituency. Thus, this reevaluation of Identifier_Type assignments is intended to improve the usefulness of the classification, but not to force implementations to deviate from any established practice that works for them.

Note also that the Identifier_Type and Identifier_Status properties do not fall under stability policies. They evolve as the use of writing systems changes and as Unicode receives feedback on that use. By contrast, properties like ID_Start and XID_Continue, which are used for *detecting* identifiers, are stable, and are not affected by these changes.