Learning More about Unicode

At the time of this writing, the official reference to Unicode is The Unicode Standard, Version 5.2, released in October 2009. If you want to explore Unicode in greater depth than the book you are now reading provides, you can start with the Unicode website, www.unicode.org. The full text of the standard is on line in PDF form; there are also many FAQs and other useful resources. In the past, the Unicode Consortium produced a printed version of the standard with every major release. You can purchase version 5.0 or borrow it from a library and check the web site for changes in the interim versions. Beginning with version 6.0, the standard will be exclusively on line (although a print on demand version may be made available).

The book *Unicode Explained* by Jukka Korpela is very good, although now slightly out of date. The earlier introduction to Unicode by Graham is also well written, but a great deal has changed since its publication. Books such as these can contain some information that is not appropriate to include in the official standard.

1.4 The Unicode Model: Characters not Glyphs

A fundamental principle of Unicode is *to encode characters not glyphs.* The Latin letter 'a' is a character that may take many different shapes: **a**, ***a***, a, *a*, and so forth. The various shapes or outlines of 'a' are referred to as glyphs, while the underlying abstraction — the Platonic Idea of 'a,' so to speak — is the character. Unicode could not separately encode an italic 'a,' an uncial 'a,' a script 'a,' and so forth; there aren't enough codepoints and doing so would make things much more complicated than they should be. It is understood that the shape of glyphs will vary from one font to another, and Unicode does not prescribe this sort of thing. Another way of expressing this is to describe Unicode as a plain text encoding. If you have ever used an email program that does not allow the use of different typefaces, font sizes, colors, etc., you have worked with plain text. The opposite of plain text is fancy text or rich text.

The distinction between characters and glyphs is one of the most basic elements of Unicode and must be understood by people who need unusual characters such as those that many scholars use in their work.

This model does pose some problems for scholarly users, however. Take the case of the symbol for the sestertius, a Roman coin. Most often this appears as ~~HS~~, but is also found as II with a horizontal bar, as IS with horizontal bar or even S with a horizontal bar.[3] I proposed

Example: the sestertius.

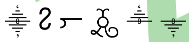
[3] The use of the letters HS as an abbreviation for sestertius is a modern printer's hack to make up for the lack of a properly shaped character in the font. Many generations of Latin students have wondered what in the world the letter H had to do with the sestertius. The sign is simply an abbreviation for two and a half (*semis* = one half), since a sestertius was worth 2.5 *asses*.

this character, among others needed by epigraphers, for inclusion in Unicode. The character was accepted, but only as one entity; that is, the *character* sestertius was accepted, but five separate glyphs were not. Had I proposed five separate characters, corresponding to the five glyphs mentioned here, the proposal would have been rejected. But scholars frequently wish to preserve information about the exact shapes of characters when they prepare editions. There are some solutions, such as adding higher-level markup or using the Private Use Area, about which more will be said below, but the fact remains that sometimes the Unicode character-not-glyph model is not ideal for scholars although it works well enough in everyday use with modern languages.

Characters with very different variant shapes.

Some characters of interest to scholars have a number of glyph variants, sometimes with quite different shapes. The first such one to be encoded was (as far as I can tell) U+ 2E0E EDITORIAL CORONIS, a mark used by ancient Greek scholars. When the Thesaurus Linguae Graecae proposed this character for Unicode, they identified five common variants and several rare ones. Later on, in connection with proposed characters such as the Roman centurial sign, some members of the Unicode Technical Committee raised questions about whether it was appropriate to encode characters that have glyph variants with widely differing shapes.

Here is the reference shape of U+2E0E EDITORIAL CORONIS followed by the five common variants:



In addition to pointing out the precedent of U+2E0E, I argued that the centurial sign and similar characters should be encoded for three reasons. First, the varying shapes would never be accepted as separate characters, leaving the centurial sign forever unstandardized. Second, having a single codepoint provides simplicity (“What’s the Unicode value for the centurial sign?” has only one answer) and searchability. Finally, the disparate forms of the centurial sign all seem to have evolved palaeographically from the reversed C shape, sometimes with a horizontal stroke to mark an abbreviation. (This is not necessarily the case for all characters with a variety of variant shapes, of course.) The centurial sign was accepted and this precedent has been set.

Encoding versus displaying texts.

This might be a good time to point out the essential difference between *encoding* a text versus *displaying* (on a computer screen) or *printing* it. If a certain language requires the letter ‘z’ with a macron, texts in this language can be encoded perfectly using the letter ‘z’ followed by a combining macron. The fact that mainstream software has not handled combining marks well until very recently (if yet) has caused problems for users of this language and has given rise to complaints that Unicode does not support their language or discriminates against it. In a perfect world it would be easy to display all needed combinations of letters and diacritics, but the fact remains that ‘z’ plus a combining macron represents the author's intent without any ambiguity whatsoever; the text is encoded correctly. Similar issues arise with higher-level markup, such as electronic editions created using markup language like the TEI guidelines. An electronic edition can preserve, for instance, information of interest to a palaeographer or epigrapher. Since Unicode does not encode glyph variants, additional steps are needed to display such information. See page **??** for some additional information about this issue.

Why some formatted characters exist in Unicode — and when (not) to use them!

Unicode does contain some characters that seem to fly in the face of the character/glyph model. For instance, small capitals are normally used as design elements to emphasize or set off a bit of text. They look good but their omission would not change the meaning of the text. Unicode does contain some small capitals, however. Some of these belong with the letters of the International Phonetic Alphabetic (IPA). In IPA transcription, small capitals indicate certain pronunciations; regular capitals or small letters could not be substituted without altering the accuracy of the transcription. Some additional small capitals were added at the request of medieval scholars, since in Old Icelandic small capitals are used to indicate doubled consonants. In Plane 1, there is a large group of Latin letters that are boldfaced, italicized, or specified with other variations. These are intended for use in mathematics, where a boldfaced 'X' or italicized 'X' may mean something very different than a plain one, and the omission of formatting would cause the writer's intent to be obscured. *Never use these "formatted" Unicode characters unless they are required to convey a specific phonemic or structural meaning.* Use your word processor's or page layout program's commands to make the document clear and attractive through the use of rich text, reserving Unicode formatted characters for situations where they must be used to preserve the meaning of the text.

1.5 Unicode and Historical Scripts

Unicode has always been willing to support historical scripts (it is, after all, the Universal Character Set), although in the beginning more attention was naturally paid to those currently in use. By now, though, a large number of historical scripts has been encoded, mostly in Plane 1; the Roadmaps at http://www.unicode.org/roadmaps/bmp/ and http://www.unicode.org/roadmaps/smp/ offer a convenient overview.

Three reasons to encode historical scripts.

People sometimes ask "Why bother encoding Lydian (or some other obscure historical script)?" The answer has three parts.

First, people are using these scripts for research — even if in very small numbers. Many of us know how difficult it was to use Greek or Cyrillic or Hebrew in pre-Unicode days. Having other historical scripts encoded extends the same benefits of standardization and easy exchange of data to other users.

Some new friends in Unicode 5.2, as shown in George Douros's Gardiner font:

U+1305F	𓁟
U+13153	𓅓
U+1330D	𓌍

Second, having the script encoded opens the way for further development. Unicode 5.2, for instance, added a basic set of Egyptian hieroglyphs based on Gardiner's Egyptian grammar. At the time this was written, there were no applications that could use them; but, of course, it was not possible to develop applications to display Unicode hierglyphs when the characters were not available. Now that the characters are encoded, things can move forward. Implementing Unicode hieroglyphics, by the way, will make a very interesting study for those who are interested in script and font issues: they were written both left to right and right to left as well as vertically; in addition, there are special needs such as the ability to enclose names in a cartouche.

Will it always be possible to add characters to the UCS?

Third, Unicode is clearly the way of the future; almost all operating systems and significant pieces of software are now Unicode-based. Those who want to use a script for scholarly purposes are better off getting on board sooner than later. This raises the question of whether it will ever be too late, i.e., whether the standard will be closed at some point. I think that unlikely, given the fact that human languages, along with their written representations, are always changing and evolving. (Some software companies would like to see the Standard finished, because that would make their work easier.) More likely, perhaps, will be a reduction in resources. The Unicode Consortium is funded by software companies, governments, and other donors. If they reduce their contributions, for whatever reason, it will become more difficult and take longer to get characters or entire new scripts added. So better to do it now, while resources are available. The Script Encoding Initiative maintains a list of scripts awaiting encoding at http://www.linguistics.berkeley.edu/sei/list.html. See also the sidebar "How Do Characters Get Into Unicode?" on page 14.

1.6 The Issue of Precomposed Characters

Base Letters and Combining Diacritics

The original intention in Unicode was to encode any base character with diacritic(s) as a sequence of separate characters (e.g., the letter 'e' followed by a macron rather than as one combined unit consisting of 'e' with a macron over it). A combination of base character plus diacritic is referred to as a precomposed character, while the sequence of two separate characters is referred to as decomposed. Individual diacritics that are designed to be placed over a base character are called combining marks and are found in the Combining Diacritical Marks and Combining Diacritical Marks Supplement ranges of Unicode.

When combining diacritics need to be shown by themselves, as in documentation such as this book, the convention is to print them over a dotted circle to show that they are not characters intended for use on their own. A dotted circle may also appear on screen if you accidentally enter a combining diacritic without an appropriate base character preceding it; this behavior is somewhat font-dependent. The Standard specifies (§2.11 and 7.9) that combining marks can be placed over a NON-BREAKING SPACE, U+00A0, to display them in isolation. This works but does not help to distinguish, e.g., the combining grave (U+0300) from the spacing grave (U+0060), nor does it give the reader a sense of how the combining mark is positioned relative to the base. For these reasons I prefer the dotted circle.

a + ◌̆ + ◌̣ = ặ

e + ◌̄ + ◌́ = ḗ

Figure 1.2: Base characters with combining diacritics. Note the dotted circle used as a base for the combining marks.

Unicode specifies that multiple diacritics should be stacked with the first one next to the base character, the second one above the first, and so on, as shown in Figure 1.2. Some exceptions to this rule are made for language-specific needs, such as in Greek when a breathing mark and an accent are placed side by side over a vowel.

Why Precomposed Combinations Exist

Avoiding precomposed characters dramatically reduces the number of codepoints that need to be used. (A codepoint is a slot, identified by number, into which a Unicode character is assigned.) As the standard was developed, however, a large number of precomposed characters were included, mainly to insure that text could be converted correctly from various existing national standards into Unicode and back again into the original encoding. In addition to the standard Windows character set, whose arrangement is mostly followed by Unicode for the first 256 codepoints, there are three additional blocks of Latin characters, many of which are precomposed combinations: Latin Extended-A, Extended-B, and Extended Additional. Characters of interest to scholars are scattered throughout these blocks. The decision was made that after Unicode 3.0 was released, no additional precomposed combinations would be added. (This was done because of the increasing reliance on Unicode by web applications and other software that change decomposed into precomposed forms in order to display text properly; such applications would have to be constantly updated if additional precomposed combinations continued to be added.) Therefore the more recent Latin Extended-C and Latin Extended-D blocks do not contain any precomposed combinations.

The first edition of this book contained several tables with various selections of Unicode characters, including one for combining marks. Such tables are omitted from this edition because the code charts are now all available online. Start at http://www.unicode.org/charts/. Studying these charts is an excellent way to become better acquainted with Unicode.

How Do Characters Get into Unicode?

Proposals for new characters may be submitted by anyone — individuals, institutions, corporations, or governments. The submitter must show that the characters proposed are actually used in printed material (by including scanned copies of publications) and that no existing Unicode character is adequate. Proposals are reviewed by the Unicode Technical Committee (UTC). If approved, they are submitted to WG3, the working group that considers characters for inclusion in ISO-10646. If WG3 accepts them, they are placed in the formal balloting process to be ratified by ISO member countries. (The process can work the other way, beginning with WG3 and then the UTC.) Once final approval has been granted by both bodies, the characters become an official part of the standard; the author of the proposal must provide an appropriate font for printing the characters.

Two groups that have taken the lead in adding scholarly characters to Unicode are the Thesaurus Linguae Graecae, http://www.tlg.uci.edu/, and the Medieval Unicode Font Initiative, http://www.mufi.info/. The TLG analyzed its extensive database of ancient Greek texts, identified some characters there that had no Unicode equivalent, and made proposals which were ultimately accepted. Not all characters are candidates for encoding, however. The TLG found that a few items in its database were exceedingly rare or not clearly understood, while others could be considered glyph variants of more common characters; these were not proposed. See the TLG Beta to Unicode quick reference guide at http://www.tlg.uci.edu/encoding/quickbeta.pdf. Those who wish to look at actual proposals can see the TLG proposals archived or at http://repositories.cdlib.org/tlg/unicode/ or proposals for Latin epigraphic characters at http://www.scholarsfonts.net/latnprop.html. MUFI has done a similar job collecting and organizing characters from many medieval sources, and its work has lead to a large number of characters added to Unicode. Even individuals can successfully propose characters, if they make the effort to provide documentation that shows the characters in use and to explain why no existing characters will work. The author of this book successfully proposed a number of characters needed for Latin epigraphy.

For information about characters that are under consideration for inclusion in Unicode, see http://www.unicode.org/alloc/Pipeline.html. A useful resource for scholars contemplating a proposal for new characters is the Script Encoding Initiative (SEI), organized by Dr. Deborah Anderson at the University of California at Berkeley. SEI keeps track of unencoded scripts and proposals that are being considered; it is also a source of assistance for scholars who are not Unicode experts as they prepare proposals. See SEI's web page at http://linguistics.berkeley.edu/~dwanders/.