

Living with Dead Languages: Greek and Hebrew for Computerised Theologians

Pre-Computer Solutions

In the old days we had just four choices if we wanted to include Greek or Hebrew in a type-written essay:

- Transliterate.
- Leave blank spaces and write in the Greek/Hebrew by hand.
- Buy a separate typewriter for each language.
- By a “golf ball” typewriter (e.g., IBM Selectric, introduced in 1961), with removable typeballs for each font desired (which also allowed for italics, etc.).



Pre-Unicode Solutions

Dedicated Software

The following examples are from my own personal history, and are not meant to be exhaustive of the many programmes that were out there “in the old days”. The most famous (and expensive) was Nota Bene.

Multi-lingual Scholar

- The programme with which I wrote my M.Div. treatise in the CLTS library in 1989. Sadly—and to the point—the programme is no longer available, and my files (which were once stored on 5 ¼” floppies) are no longer accessible. Even if I had the disks, the data would be unreadable. (The file structure involved add a “language” byte in front of every single character of the text.)

ChiWriter/MegaWriter

- This programme was designed for mathematical equations and complex languages. It had the ability to subdivide a line of text into multiple levels, allowing accents to be placed above words and vowel points underneath.
- This is the programme I used to write my S.T.M. thesis in 1992. Again, the programme is no longer available (after upgrading to Windows 95 I was no longer able to get it to install and run), nor are my files any longer readable. As the files are ASCII (rather than binary), I can view them with a text editor, but it would take an enormous amount of work to make them publishable. Fortunately I had the sense to export the files to text only when I was last able to run the programme, but the Greek and Hebrew portions are unreadable. I have therefore scanned the printed text into a pdf for web distribution. See [www.brocku.ca/concordiaseminary/winger/Winger_The_Priesthood_of_the_Baptized_\(STM\).pdf](http://www.brocku.ca/concordiaseminary/winger/Winger_The_Priesthood_of_the_Baptized_(STM).pdf).

Dedicated Fonts

TrueType Fonts under Windows

- The principles discussed here apply to any font technology used on any computer system, but I confine myself to .ttf and Windows, with which I have experience.
- Like DOS, Windows began with a simple 128 character set (ASCII), basically covering the computer keyboard, with variants depending on the local language installed (called code pages). An “extended” range brought the number of characters available to 256, and allowed for the accents and special characters used in modern Western European languages. The extended characters were available by holding ALT and typing a three-digit code on the number pad. Later Windows code pages were accessed with the same technique, but by placing a zero in front. Examples:

DOS-style (code page specific)	Windows style (input language specific)
ü 129	ü 0252
é 130	é 0233
ä 132	ä 0228
à 133	à 0224
ç 135	ç 0231
ê 136	ê 0234
è 138	è 0232

- Unfortunately, this system did not provide enough code points for all the characters needed in more exotic languages. Microsoft and Apple provided different “code pages” that contained a different set of 256 characters depending on which language was installed on the computer system, but only one code page could be used at a time.
- The early solution was to create special fonts that used the same 256 points as the installed code page, but changed the look of each letter so that it *seemed* like it was a different language. In this way the Latin alphabet was made to coincide with the Greek alphabet (for example). You typed a “b” and you got a β. Thus, as far as Windows was concerned, the Greek font was simply a fancy Latin font.
- So, one would type *logos*, and it would display in WinGreek as λογος.
- The problem was that each font designer would make different decisions about where to place (1) the odd Greek characters (like final sigma in κρισις, or the letter ψ); and (2) the proper keyboard and character equivalents for the accent marks (should a smooth breathing correspond to an apostrophe, a comma, or a round bracket?).
- The resulting problem was that the keystrokes that would produce the correct Greek characters in one font would lead to significant errors when the text was converted to a different font. This would happen when sending a document to someone who didn’t have the font you used, or when a given font was lost or no longer functioned. Note what happens when the keystrokes appropriate to the BibleWorks are used with other fonts:

<i>type o` lo,goj and get:</i>	
ó λόγος	bwgrkl (BibleWorks)

ο´ λο,γος	Greek (WinGreek)
ὀ λο,γὸ	GraecaII
οῶ λο,γος	SPIonic

- Part of the problem was that these fonts took two different approaches to accent marks:
 - Some fonts (e.g. WinGreek) created **combined characters**, such that each of the following consists of just one character: ᾱ ᾶ ᾷ Ᾱ Ᾰ. (*move the cursor across these letters to see*)
 - Other fonts (e.g. BibleWorks) created accent marks that were set with **negative character width** so that they would “back up” and land on top of the previously-typed vowel: ᾱ ᾶ ᾷ Ᾱ Ᾰ.
- *If you use such fonts, take note of the following warnings:*
 - (1) *Archive copies of any Greek or Hebrew fonts you have used so that when you upgrade to a new computer or operating system you will be able to reinstall them easily and retain access to your old files.*
 - (2) *When sending a file to someone who might not have the necessary font, choose the option to “encapsulate [embed] fonts” in the saved file.*
 - (3) *Create pdf versions of important files, and remember to encapsulate fonts in the pdf.*
- Hebrew caused particular problems because of the excessive number of vowel marks and other diacriticals, together with the fact that it needed to be typed right to left. In addition, the placement of the vowel points changes depending on the width of the Hebrew letter. Prior to the advent of fonts using negative character width (available now in, e.g., bwhebb [BibleWorks]), I used a system with an MS Word macro that created a *field code* to back up and over-type two letters, as well as a macro that enabled right-to-left typing. Needless to say, this hasn’t aged well as Word went through upgrades. Nor can these files be imported into any other word processor. Note how one excerpt from my Th.D. dissertation now looks:

When Naaman appears in Israel’s royal court, he immediately speaks (לִּימְנָאן) the contents of the letter (:6), after which King Ahab reads it aloud (בִּקְרָא) for himself (:7). The expression for “literacy” is also instructive: יָדַעַתְּ “to know (a) writing.” In Is. 29:11-12 it is the competence required to answer the command קְרָא “read!”

Beta code

- Developed in the 1970s for the *Thesaurus Linguae Graecae*, a project to digitise all ancient Greek and Latin texts, beta code was an attempt to standardise the mapping of Greek characters to the ASCII codes of the Latin alphabet, and to provide codes for an extended character set. Although today Unicode is the standard, beta code remains significant because the *TLG* is still the most significant classical database (www.tlg.uci.edu/). This is a subscription service available through Brock’s library. There is a limited free database more easily accessible through the Perseus project (www.perseus.tufts.edu).
- Beta code takes a simple sequence of ASCII characters like this:

pa=sa te/xnh kai\ pa=sa me/qodos, o(moi/ws de\ pra=ci/s te kai\ proai/resis

and translates it into a variety of Greek formats such as this:

πᾶσα τέχνη καὶ πᾶσα μέθοδος, ὁμοίως δὲ πράξις τε καὶ προαίρεσις

- Nestle-Aland editorial signs are available in U2E00-U2E0D; letter forms are in the Supplementary Multilingual Plane (e.g., Ϳ U1D50E; ͽ U1D50F; ; U1D510; Ϳ U1D513; ͽ U1D52D; Ϳ U+1D459)
- The problem with Unicode is how to access this large number of characters from a tiny keyboard. One solution is to type in a code (like the old ASCII days). For help see http://www.fileformat.info/tip/microsoft/enter_unicode.htm. Essentially, these are the options:
 - 1) Hold the ALT key, press + on the number pad, then enter the hexadecimal code. On some systems (XP) this requires a registry edit to make it work.
 - 2) In some Microsoft programmes (MS Word, Wordpad) you can type the hexadecimal code in your text and then press ALT-X to convert it (this works in both directions).
 - 3) Various character maps can be used to cut and paste the character you want (e.g. “Insert Symbol” or Babelmap).
 - 4) There’s a dedicated website: <http://users.ox.ac.uk/~tayl0010/polytonic-greek-inputter.html> that will help you type Unicode Greek and then cut and paste it.
- Clearly none of these options is viable for entering large amounts of biblical text. One solution is simply to cut and paste text from BibleWorks (or Logos). BibleWorks can be configured to export text either with its proprietary font (old-fashioned) or in Unicode (new-fangled). Logos works completely with Unicode. There are also plenty of commercial packages available to help you type Greek (e.g., GreekKeys, Nanos, Antioch, Multikey, Thessalonica, Linguist’s Software). A very good solution (it’s free!) is a utility from Logos that displays a keyboard on screen with an editing box and lets you point and click various ancient languages: <http://www.logos.com/shibboleth>.
- The best solution is to install additional languages and keyboards for Greek and Hebrew. There are commercial products available like Tavultesoft Keyman, but the free keyboards that ship with Windows work perfectly fine.
- For Greek you need to install the “Polytonic Greek” keyboard, since modern Greek only uses one accent (monotonic). The keyboard will now enter the correct Unicode character, and will combine accents and vowels into the correct “pre-composed” character.



- The accent and breathing marks are on “dead keys” that must be pressed before the vowel. (Note, however, that should a keyboard come available that allows for the use of “combining diacriticals”, these will always be typed **after** the vowel.)
- An alternative to the Microsoft Greek keyboard is provided by Logos. The Logos keyboard uses combining diacriticals rather than precomposed characters. In other words, the vowel and its

accent remain separate characters. This is preferable for editing and searching, but, depending on the font, may produce inferior looking printout. See <http://www.logos.com/support/downloads/keyboards>.

- Since there are dozens of large Unicode fonts available today, it is no longer necessary to recommend specific fonts for Greek (or Hebrew). However, some are more capable than others. You may wish to choose a single Unicode font to use for both English and Greek, so that they look good together.
 - The following fonts come free with Windows and have large, capable Greek and Hebrew blocks: Arial Unicode MS, Times New Roman, Palatino Linotype (no Hebrew), Lucida Sans Unicode, and Tahoma.
 - More elegant and capable fonts are available for free. I heartily recommend: Cardo, Linux Libertine, Linux Biolinum, Gentium, SBL Greek, and SBL Hebrew.
 - An up-to-date list is available at <http://www.tlg.uci.edu/help/UnicodeTest.php>.
 - For a huge variety of historic Greek fonts, see <http://www.greekfontsociety.gr>.
- For Hebrew, things are slightly more complicated. Windows does not include a keyboard driver that will provide all the necessary vowels and accents. The best solution is available from the Society for Biblical Literature: www.sbl-site.org/educational/BiblicalFonts_SBLHebrew.aspx. This site has the best instructions for installing multilingual keyboards on both Windows and Mac. (Logos also provides a biblical Hebrew keyboard at <http://www.logos.com/support/downloads/keyboards>.)
- The complexities of typing Hebrew, including RTL typing and the placing of vowel points and accents, means that it is not well supported among word processors. On Windows, MS Word does the best job. But for those on a budget, OpenOffice will do nearly as well, once complex scripts are properly enabled.
- Note that once you have installed Greek and Hebrew as languages and keyboards, you switch between them using the “language bar” (with the shortcut ALT-SHFT). Most word processors will then automatically change the language setting (for spell checking), and switch to right-left entry for Hebrew.
- *The beauty of using Unicode is that any large Unicode font (i.e., that contains all necessary characters) will display the Greek and Hebrew correctly.* This means that you can change fonts without messing anything up. Note the following examples:
 - אָהָם Arial Unicode MS
 - אָהָם Lucida Sans Unicode
 - אָהָם Tahoma
 - אָהָם Times New Roman
 - אָהָם Cardo (<http://scholarsfonts.net>)
 - אָהָם Linux Libertine (www.linuxlibertine.org)
 - אָהָם SBL Hebrew (see above)
- While any of the above fonts will render Greek and Hebrew well, many experts recommend using the new (and free) SBL Hebrew font because it contains every conceivable Hebrew feature

(even for things like the Dead Sea Scrolls), and places the marks very well. Note that Word allows you to set a specific font for use with “Complex scripts”.

- Incidentally, for typing French, German, Latin diacriticals, etc., it is worth installing the US Multilingual Keyboard (or UK Extended, which I prefer):

~ `	! 1 1	@ 2 2	# 3 3	\$ £ % 4	% 5	^ 6	& 7	* 8	() 9	' 0	- = -	+ ÷ =	← Backspace
Tab ↔	Q Ä	W Å	E É	R ®	T Þ	Y Ü	U Ú	I Í	O Ó	P Ö	{ } []	 \	! ~
Caps Lock ⬆	A Á	S Š	D Đ	F	G	H	J	K	L Ø	: °	" "	Enter ↵	
Shift ⬆	Z Æ	X	C ç	V	B	N Ñ	M	< Ç	>	? /	Shift ⬆		
Ctrl	Win Key	Alt							Alt Gr	Win Key	Menu	Ctrl	

- There are a number of options available for converting the Greek in older documents type with specialised fonts into Unicode:
 - <http://www.users.dircon.co.uk/~hancock/antioch.htm> (commercial)
 - <http://www.greektranscoder.org/> (free)
 - <http://litot.es/unicode-converter/> (free)
 - There are a few Word macros that will convert BibleWorks fonts into Unicode. One is provided with BibleWorks (unimacros.txt), but there are better ones available through their online forum. E-mail me (twinger@brocku.ca) if you need help with this issue.

Resources

PERRY, DAVID J., *Document Processing for Classical Languages*, 2nd ed. (Greentop Publishing, 2010). See <http://scholarsfonts.net/docproc>. Available in the CLTS library.

Thomas M. Winger
15 February 2011