[**GreekTranscoder**](http://www.greektranscoder.org) **v. 1.0.8  
Conversion program for polytonic Greek text encodings.  
Copyright (c) 2005-16 David-Artur Daix, <**[**mailto:daix@ens.fr**](mailto:daix@ens.fr)**>  
École Normale Supérieure, Paris, France.**

Introduction: 1

System requirements: 2

Installation: 2

Upgrading GreekTranscoder: 3

Adding GreekTranscoder to the Ribbon : 4

How GreekTranscoder Works: 5

Performance tips: 7

Customization: 8

Technical notes: 8

Troubleshooting: 10

Acknowledgements: 11

Contact: 11

# Introduction:

GreekTranscoder is a program which converts polytonic Greek characters written using one text encoding into another one. Its primary goal is to allow the conversion of documents using older fonts and encodings into Unicode fonts. However, it also allows converting text between older encodings as well as from Unicode into those obsolete formats.

Currently GreekTranscoder supports the following legacy encodings:

* Beta Code
* GreekKeys
* Ismini
* LaserGreek[[1]](#footnote-1)
* Paulina Greek
* SGreek[[2]](#footnote-2)
* SPIonic
* SuperGreek[[3]](#footnote-3)
* Vilnius University
* WinGreek (and Son of WinGreek)

And, of course, GreekTranscoder supports Unicode, using either composed or composing characters. The program lets you choose which option you wish to apply to your Unicode text. It also lets you conform to the TLG usage regarding Unicode codepoints which, according to the TLG, should be “deprecated”. A Unicode text can be converted into Unicode itself with different conversion settings.

**Note to Windows users :** in the past, one had to use “symbol” fonts on Windows to type ancient Greek. Making a symbol font was a way of getting round the problem of Windows excluding eight code points from standard text fonts, when these were needed because the classical Greek character set is so large. This problem never affected Macintosh computers. However, in Windows from Word 97 on[[4]](#footnote-4), “symbol” fonts were treated differently from “text” fonts: the characters were no longer considered as letters, but as symbols, which meant that breaks at the ends of lines might occur in mid-word for instance. These old fonts should never be used on a modern computer, on any platform. Text written with a symbol font and viewed in a text font, or vice versa, will appear as a lot of little blank squares. GreekTranscoder does not support those obsolete fonts and has no way of identifying them (quite often older “symbol” fonts and newer “text” versions of them bear the same names). So you should make sure you’re using current versions of your Greek fonts.

# System requirements:

* **Macintosh**: Word 2004 or later, which means Mac OS X 10.2.8 or later. Word 2004 is the first version of the application that fully supports Unicode, which is required by GreekTranscoder.
* **Windows:** Word 2000 or later (and consequently a Windows release that supports that Word version). Word 97, although it supports Unicode, uses an older and incompatible version of Visual Basic. GreekTranscoder has been tested successfully as far back as Word 2000 and Windows 98 SE.

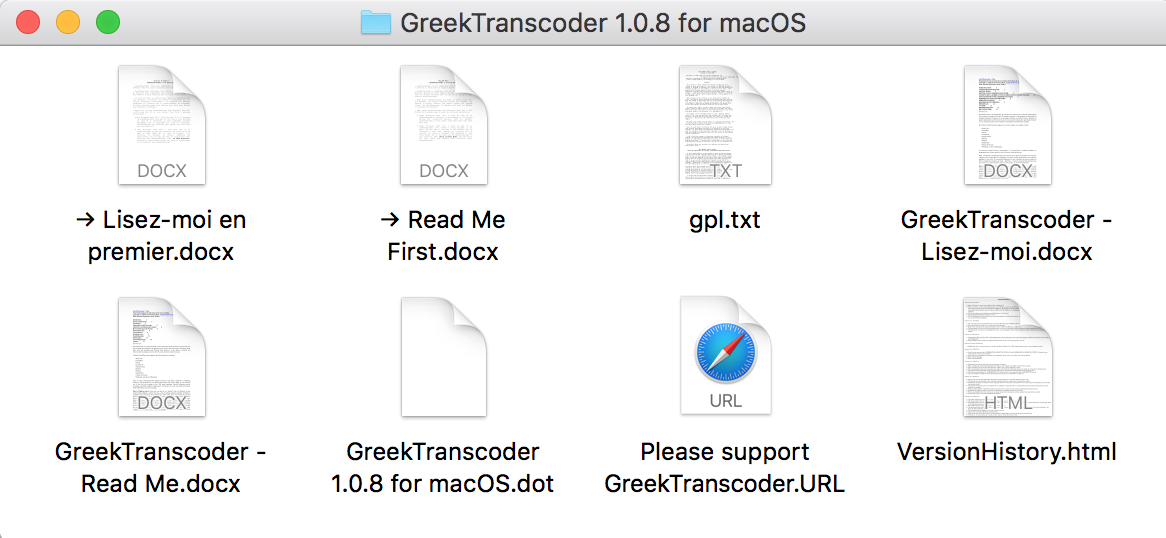
# Installation:

GreekTranscoder is distributed as a Word document template. It is programmed using Visual Basic 5 (the latest version available in the Macintosh version of Office) and is available to the user as a macro in Word

There are two versions of the program: one for macOS and one for Windows. Make sure you use the one made for your operating system.

The program is distributed under the GPL license.

You’ll also find a direct link (the “.URL” file) to the page where you can learn how to make a [donation](http://www.greektranscoder.org/#donate) to support my work on GreekTranscoder:

****

**Note:** the source code of the program is available simply by opening in Word the “GreekTranscoder.dot” file and launching Visual Basic Editor; there is no binary (compiled) version of the program; the code is interpreted at runtime.

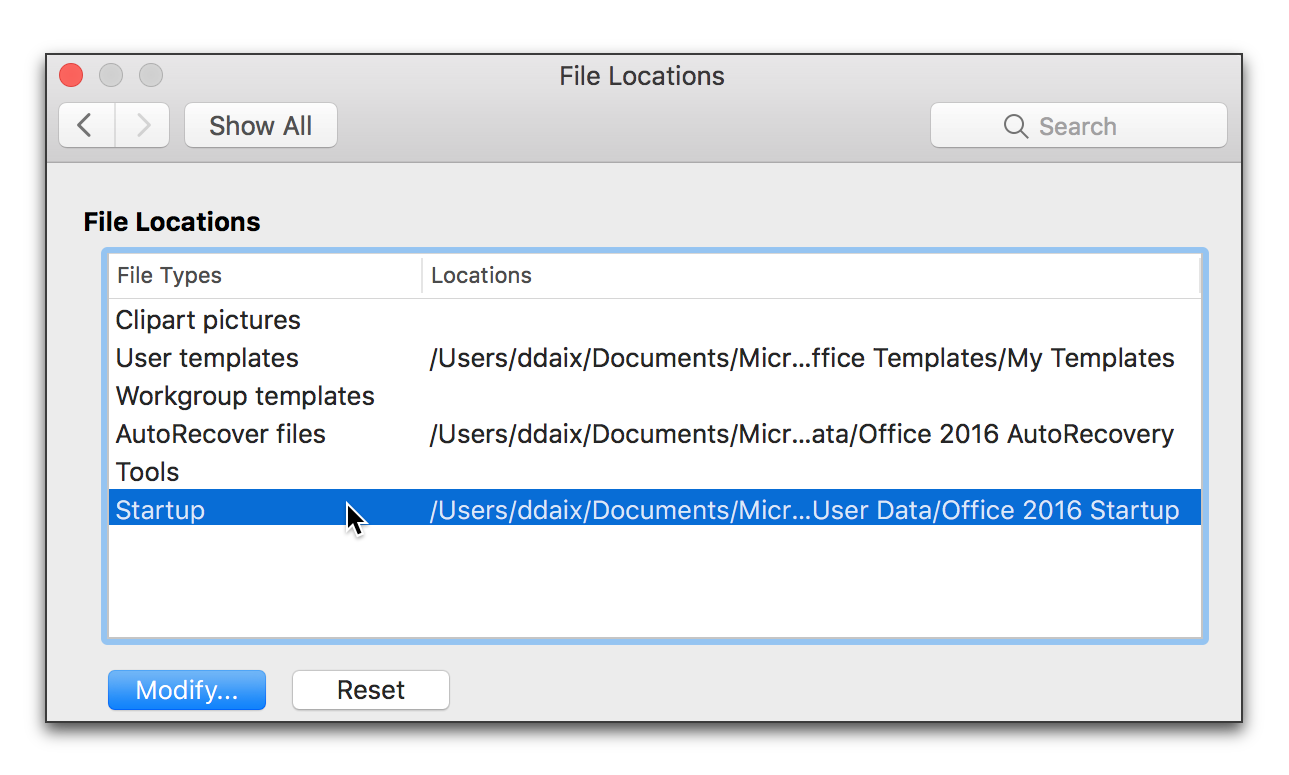
The best way to install GreekTranscoder so that it’s always available is to put the “GreekTranscoder.dot” template in your Word “Startup” folder. If you don’t know where that folder is located, go to Word preferences and in the “File Locations” section you’ll see a “Startup” item. That’s where the files should go.

**Note:** under Windows, that folder may be hard to locate. In recent versions of that operating system, the folder path is usually:

C:\Users\[your name]\AppData\Microsoft\Word\STARTUP.

However, when navigating there, the “AppData” folder is hidden and you must make it visible before you can add items to it.

Here is what the “File Locations” section looks like under Mac OS X:



Document templates located in that folder are loaded at startup as global templates and do not bring up “macro warnings” when the application is launched (I’ve created every line of code in the program myself; but you can of course inspect the code yourself as well.)

After having copied GreekTranscoder in your “Office Startup” folder, launch Word and the program will become available to all the documents you open from then on.

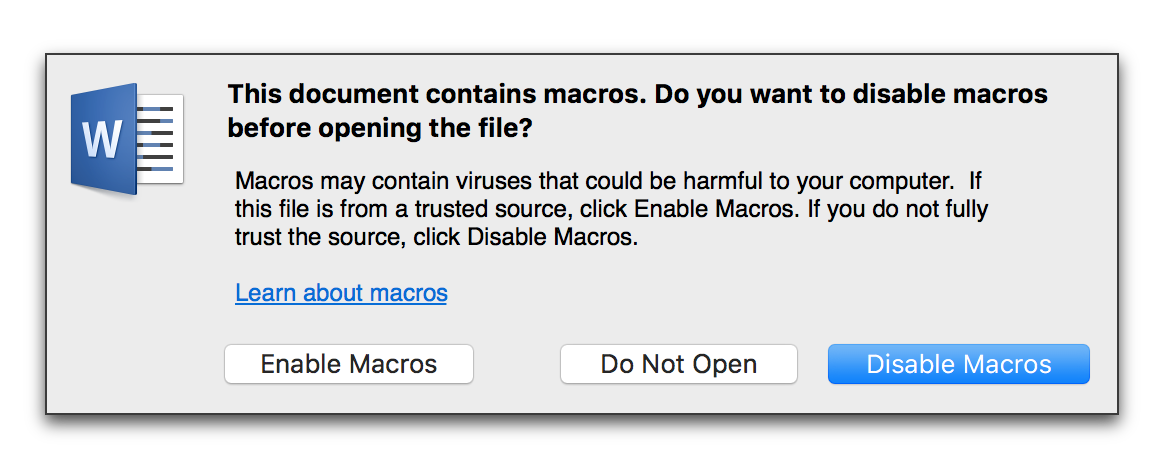
# Upgrading GreekTranscoder:

If you’re upgrading to a new version of GreekTranscoder, you must:

1. Quit Word if it’s open.
2. Delete the old version of GreekTranscoder and install the new one.
3. Launch Word again.

That’s the only way to ensure that the installation of the new version goes smoothly.

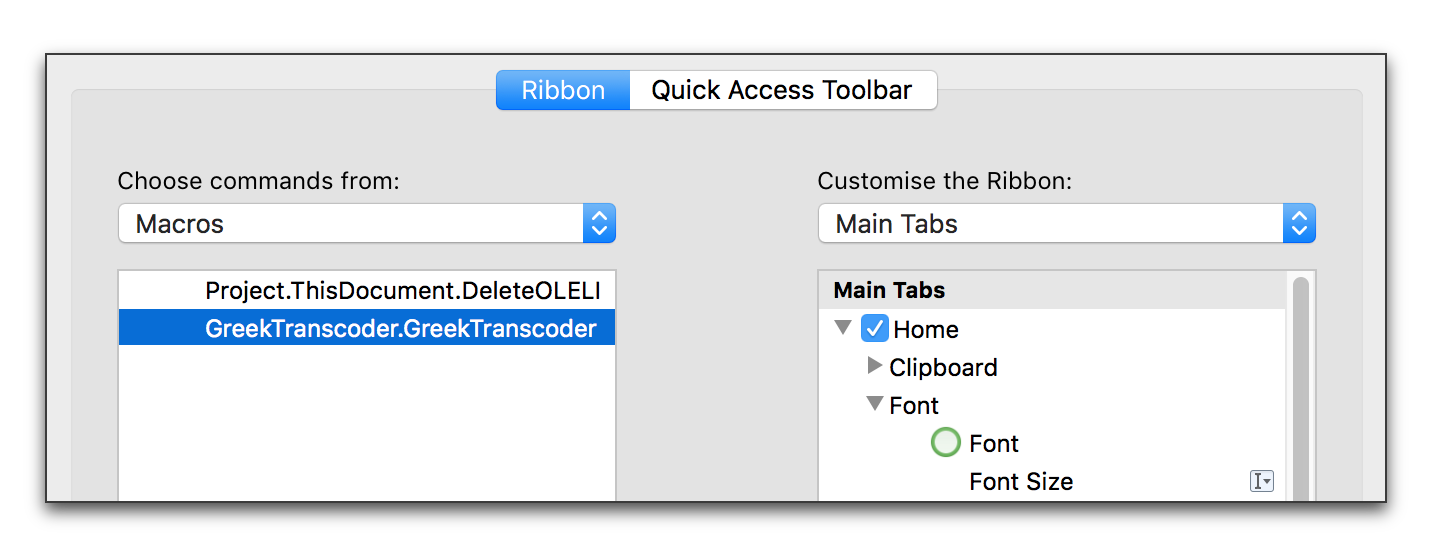
If you don’t want the program to be available all the time, you can install it anywhere you want and simply open it when you need to use it. Word will bring up a warning dialog when you launch GreekTranscoder this way:



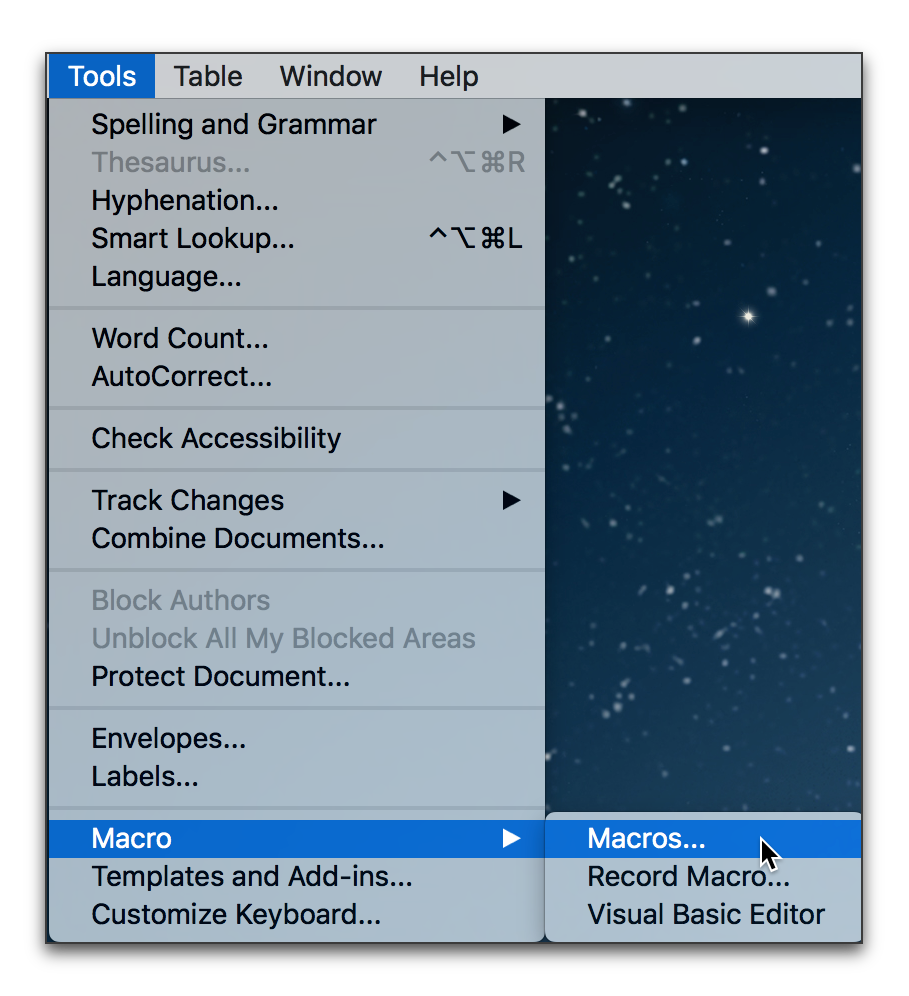
You must click on “Enable Macros” for the program to work, of course.

# Adding GreekTranscoder to the Ribbon :

To access GreekTranscoder quickly and easily, you can add it to your Ribbon. To do so, you must customize your Ribbon :

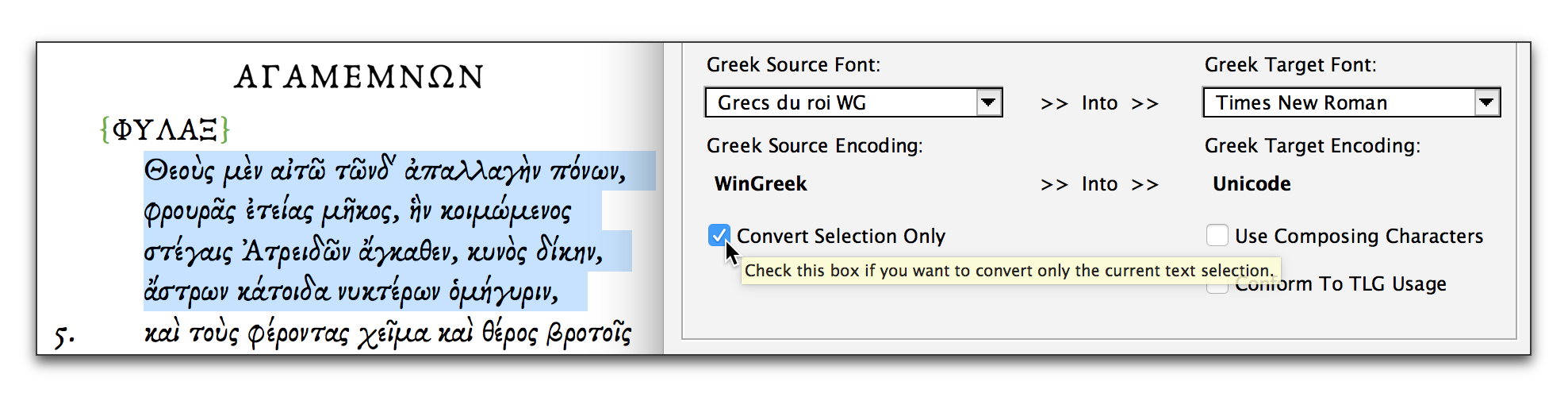


You can also launch GreekTranscoder by choosing “Macros”in the “Tools”menu :

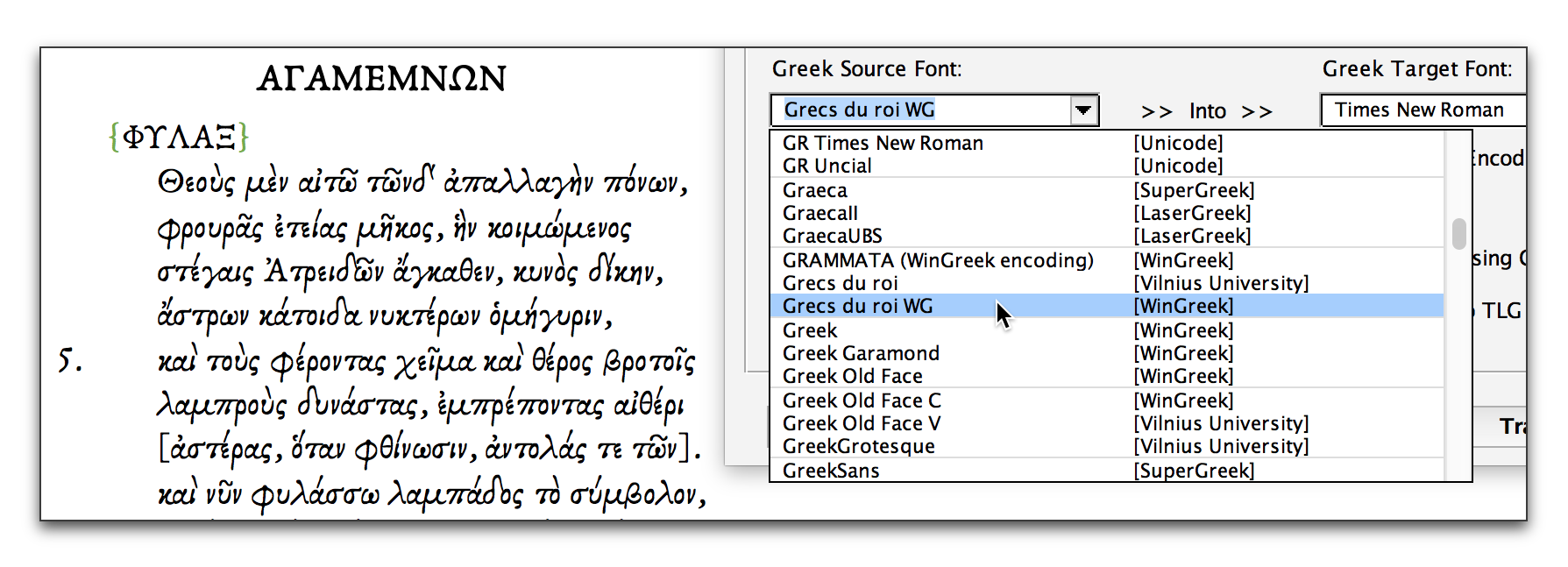


# How GreekTranscoder Works:

When you launch the program, a form appears. You can select several options, depending on what you want to do. Many are self-explanatory. All come with “tooltips”:



Word can’t identify encodings, but it can find and replace fonts based on their names, even if they’re not installed. To transcode your text, just select the source and target Greek fonts (and thus encodings) in the provided lists. The first column contains the fonts’ names; the second their encodings:



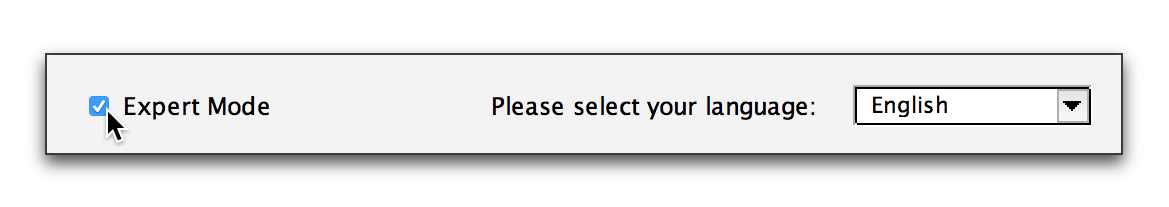
The source font is the one currently used in your document. The target font is the one that is based on the encoding you wish to use from now on.

If the fonts you select displayed problems during my testing, a message may appear to warn you and to suggest that you switch to another font that uses the same encoding.

Some fonts, while bearing common names, such as Helvetica or Times, may only support polytonic Greek in a specific version of the fonts distributed as part of a specific OS release. If you select such a font and you’re not running the right OS, you’ll be alerted to that fact.

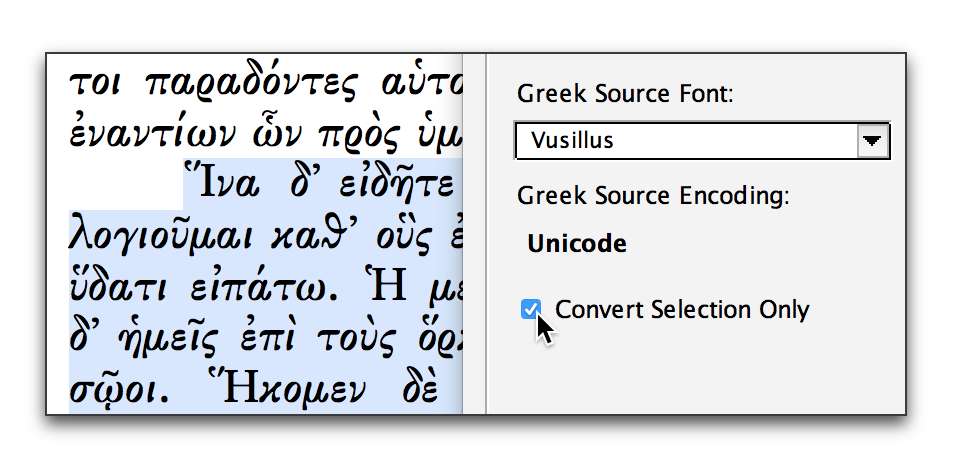
**Note**: I haven’t had the time to implement user preferences in this release, so the program won’t remember your previous settings when you launch it again; they’ll be reset instead.

If your fonts, while not listed, use one the supported encodings, switch to Expert Mode, select the correct encodings and enter the fonts’ names exactly as they appear in Word’s Font dialog so that Word and GreekTranscoder can find and process them:



**Note:** in Expert Mode, the program trusts the information which you provide to transcode the characters. It doesn’t have any way to verify whether they’re correct or not. Be careful.

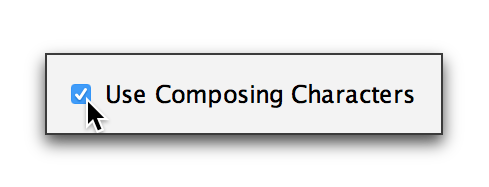
If some text is currently selected in the document, the “Check Selection Only” option will be turned on and, if the only font used in the selection is among those supported by the program, it will be set as the source font:



**Tip:** as of version 1.0.2, when a source encoding contains composing characters, GreekTranscoder automatically verifies and sorts them, i.e. makes sure they are entered in the correct order and take the correct form.

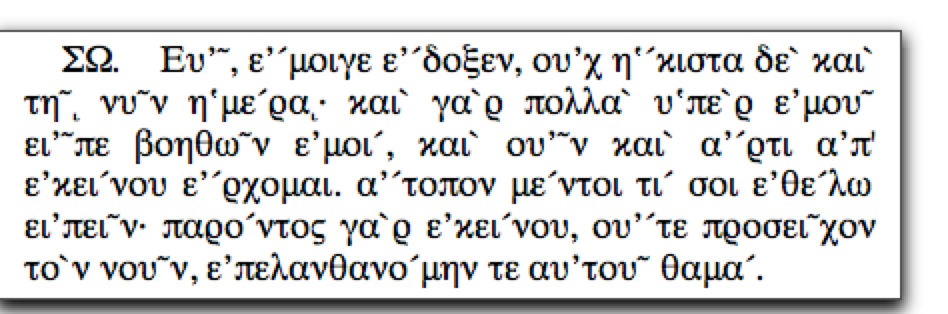
Some options apply only to Unicode fonts. They stay disabled otherwise:

* “Use Composing Characters” applies to Unicode target fonts. When typing Greek using Unicode fonts, you can either use (pre)composed characters (Normalization Form C), in which the letter and its diacritics form one unique glyph; or composing characters (Normalization Form D), which remain separate entities and must be entered in a precise order (called “canonical”) so that they can be correctly combined into the desired character (diacritics that can be used as composing characters are called “combining” or “non-spacing”). If that option is checked, composing rather than composed characters are used. If unchecked, composed rather than composing characters are used:

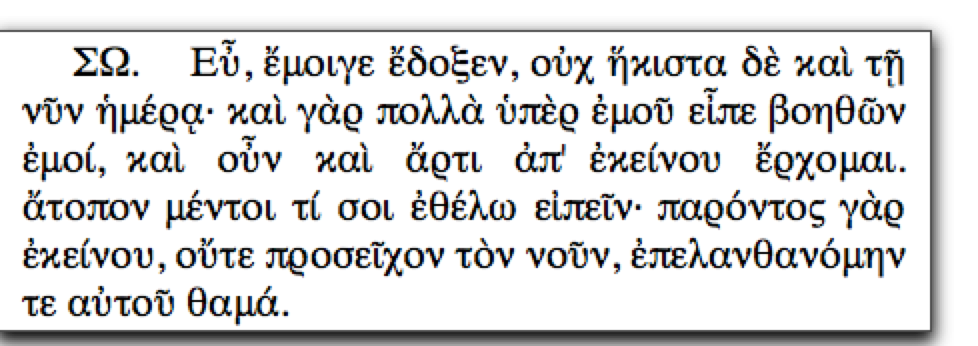


**Note:** if both the source and target fonts are Unicode fonts, depending on the setting you choose, the program will convert the composing or composed characters found in the source fonts into the target font, “normalizing” the text so that it uses only composed or only composing characters. The same will apply to the “Conform To TLG Usage” option.

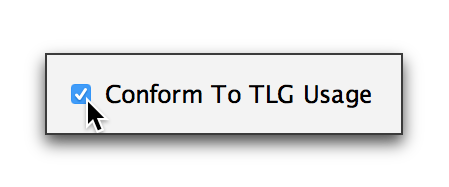
***Nota Bene*:** most applications and many Unicode fonts do not support combining diacritics correctly at this time. Here is a text sample captured in Word 2004, which does not yet displays composing characters correctly:



And here is the exact same sample captured in TextEdit, which, thanks to Mac OS X 10.4 support for composing characters, is displayed correctly:



* The TLG Consortium makes many recommendations regarding the use of Unicode characters to type ancient Greek. In particular, it deprecates those codepoints it deems redundant, preferring for instance the generic alpha with tonos also used in modern Greek to the specific alpha with oxia found in the Unicode Greek extended range. If you wish to follow those guidelines, check the “Conform to TLG usage option”:



When you click on the “Transcode” button, GreekTranscoder creates a copy your original document under the same name, but with the “.xcoded” suffix added, and save it into the same directory as the original. That way your data should always be safe.

Work on the “.xcoded” document as long as you need and during that time do not rename it and do not use GreekTranscoder again on the original document or the current “.xcoded” document will be overwritten with a fresh copy of the original. Once the conversion is done, you may rename the “.xcoded” document whatever you want.

# Performance tips:

The time needed to perform a conversion is a function of the size of the document and of the number of replacements needed.

Transcoding a text selection, even if you have selected a very large portion of text, will always be much quicker than converting the whole document, as that means checking not only the main body of text, but also headers, footers, footnotes, etc.

Also, when dealing with very large documents, you should probably convert them one piece at a time to speed up the conversion process and help Word cope with it.

Moreover, since GreekTranscoder saves the document regularly during the conversion process, another drawback of working on very large document is that those backup operations will take a while.

GreekTranscoder includes several optimizations: it switches to Normal view before proceeding, and displays hidden fields as well so that index entries and cross-references for instance are converted as well. The document’s views options will be restored to their original state afterwards.

# Customization:

This program is released under the GPL as open-source (the text of the license is included in this distribution). It means the source code is readily available: just open the template in Word as you would any other document and launch Visual Basic Editor to access it.

For instance, if you need to add the name of your favorite font, currently absent from the program (providing it uses one of the supported encodings), you can do so yourself by adding a couple of lines of codes in the “Private Sub UserForm\_Initialize()” routine:

[…]

.AddItem "Athenian"

.List(5, 1) = "[GreekKeys]"

.AddItem "Betaread"

.List(6, 1) = "[Beta Code]"

[…]

If you don’t want to renumber all the entries, simply add you font’s name and encoding at the very end of the list. You’ll also need to update the “Private Sub boxFindFont\_Change()” and the “Private Sub boxReplaceFont\_Change()” routines to include your font’s name under the right category.

That’s all there is to it.

# Technical notes:

The routines in GreekTranscoder were designed according to the specifications of each supported encodings. Unfortunately, some fonts, while based on those encodings, do not follow those specifications entirely or even sometimes correctly. That problem shouldn’t affect common characters. But if your text contains rare characters, you may notice some differences in the converted text. Short of supporting individual fonts with their idiosyncrasies rather than encodings — which is an impossible task —, there isn’t anything I can do about it. For instance, the SPIonic specifications list “v” as the equivalent of a lowercase digamma. SGRead follows the rules and inserts a small digamma. However, the font SPIonic itself stores a small letter stigma there. Whereas the author of the font Tazoatrekey has decided to insert a small letter koppa instead. In this particular case I insert a small letter stigma, as SPIonic is the standard SPIonic font, even though it does not follow its own specifications. But it’s a real problem. One more reason to switch to Unicode.

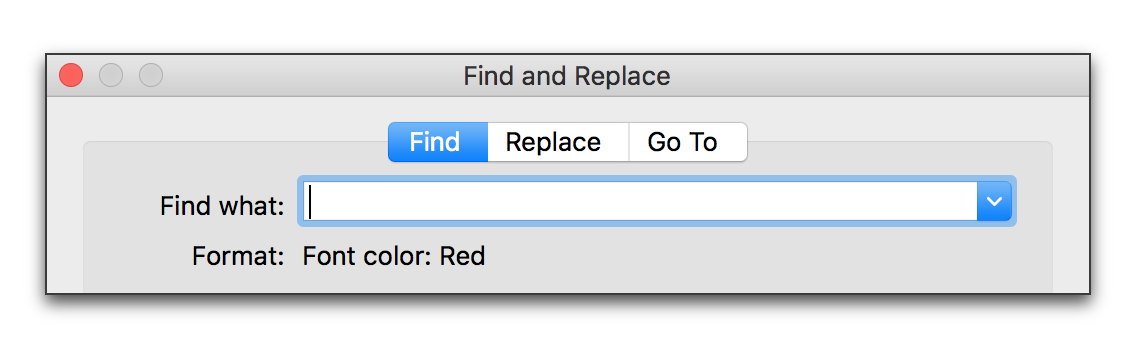
About archaic letters: most encodings only include one character for each archaic letter, and usually it’s a small letter, but not always. GreekTranscoder, unless both the small and capital letters are included in the encoding, considers the letters to be small to be able to perform more satisfying conversions. So, in the example given above, even though what remains in SPIonic is actually an uppercase digamma, the lowercase one having been replaced by a small numeral stigma, the program treats that digamma as a small letter so that it may transcode it easily, since most other encodings only have the lowercase character.

Note to SuperGreek users: over the years, Linguist’s Software has produced many versions of its fonts. The original “SuperGreek” font only existed as a Postscript Type 1 font for use on Macintosh computers. Later, TrueType fonts were created, such as Graeca. There are many problems involved in transcoding those fonts and GreekTranscoder works around most of them. However, one issue remains: normally, in the SuperGreek encoding, the tilde ‘~’ character — ASCII 126 —produces a combining tilde for use with Latin capital letters. That character was replaced in the “LaserGreek” encoding, descendant to “SuperGreek”, by a final sigma, thus offering an alternate keystroke to the regular double quote ‘"’ — ASCII 94 — used to represent that Greek character in both SuperGreek and LaserGreek fonts. Some versions of the Linguist’s Software fonts, while conforming to the SuperGreek encoding for all the other codepoints, have a Greek final sigma at ASCII 126. GreekTranscoder has no way to know which versions of those fonts you’re using and follows the encoding’s specifications. So, if you’re used to typing a tilde to insert a final sigma, before running GreekTranscoder, replace all the tildes in the text by double quotes (and make sure the “smarten quotes” options are off in Word).

When retrieving texts online, you should always ask for a Unicode text display if available. When using other encodings, the results you get may be incorrect or depend on a specific Latin text encoding (MacRoman, Western etc.). For instance, to see a text using GreekKeys in your browser, you need to select the MacRoman encoding. But to see a text using Sgreek, you must select the Western (ISO-Latin 1) encoding. Failing to do so produces faulty samples.

When using GreekTranscoder on a Macintosh, you won’t be able to use the tab key go move between fields: it’s currently a limitation of Visual Basic applications when used on that platform.

If some characters can’t be converted correctly, they are left in the original font and colored in red. You can then proceed to find them using only that formatting criteria in Word’s Find dialog:



When converting encodings’ that use composing characters, the program makes sure the character sequences follow the “canonical” order.

When producing SGreek and SPIonic texts, GreekTranscoder puts all the diacritics that precede capital letters on non-breaking spaces.

When producing SGreek texts, GreekTranscoder creates Beta Code-compatible character sequences, i.e. it does not make use of the composed non-spacing diacritics offered by Sgreek in the high-ASCII range.

SPIonic is a seemingly simple encoding, but the use of different diacritics for “wide” and “narrow” letters creates many difficulties. Even the TLG gets confused. So, when converting SPIonic, the program conforms to tradition: only alpha and omega are considered “wide” in Greek and use the diacritics meant for “wide” characters. Also, since SPIonic encourages the use of composed non-spacing diacritics, they are used instead of the usual Beta Code diacritical sequences when producing SPIonic texts.

WinGreek fonts do not include straight quotation marks: the program does its best to smarten straight quotes found in other fonts when converting a text into WinGreek. However, it may miss some, which will be colored in red if they haven’t been transcoded.

SPIonic does not contain any quotation marks at all. The font Tazoatrekei does, but since it’s an “enhancement” specific to that font, GreekTranscoder does not take those characters into account when converting to or from the SPIonic encoding.

The only official version of Ismini is the PostScript Type 1 font created by Nikos Goulandris for the Macintosh. There are Truetype clones available on the Internet, for instance for browsing the web site “Tabularium”. Those fonts are not exactly the same as the original Type 1 font. When converting a text from Ismini, the program will properly convert the characters found at different Unicode codepoints in the Type 1 and in the Truetype fonts. However, when converting a text into Ismini, it will only produce the codepoints found in the official Ismini font.

Sometimes, at least on a Mac, the program’s progress appears in Word’s status bar. Sometimes a small window comes up in the middle of the screen instead. I don’t know why that happens and I have absolutely no control over that strange Word behavior.

There are other transcoding tools available if you’re using older versions of Word on the Macintosh and do not need Unicode support, such as Daniel Béguin’s GreekFontsConverter. There is also another free tool in development, called [GreekConverter](http://members.aon.at/neuhold/antike/grkconv_en.html).

# Troubleshooting:

Working with encodings, especially badly documented ones, is difficult, to say the least. So many things can go wrong. However, GreekTranscoder has been extensively tested and should work around most issues, provided you use fonts that conform to the encodings’ specifications.

If, when running GreekTranscoder, the program refuses to transcode the document because it can’t seem to find the source font you’ve selected, whereas you know for sure the font is present, try limiting the selection to the Greek text only, or change the Greek font to a different font and then back to what it was before. This problem can occur when documents written on a Mac are transcoded on a PC, or vice versa; or when the selection contains a few characters from a font that is not installed (not necessarily the Greek font itself) and that Word doesn’t know how to handle.

If the conversion produces unexpected results, make sure the text was not written or is not displayed using an obsolete Windows “symbol” font.

If nothing works, you may send to me an e-mail and explain the problem in detail. Please join the problematic document if you can as well as the Greek font that’s causing you trouble.

# Acknowledgements:

First, many thanks to Ralph Hancock, author with Denis Liégeois of [Antioch](http://www.users.dircon.co.uk/%7Ehancock/index.htm) and many wonderful fonts, old and new. His help has been invaluable.

And many thanks to Daniel Béguin, Pierre Mounier and Samuel Provost for their support.

The transcoders’ code was generated using a database I’ve created under [FileMaker Pro 7](http://www.filemaker.com). It was the only way to incorporate changes quickly and efficiently into the program

[PopChar X](http://www.macility.com/products/popcharx/index.html) has proved an essential tool when creating transcoders.

All the screen captures in this documentation have been done using Ambrosia Software's [Snapz Pro X 2](http://www.ambrosiasw.com).

GreekTranscoder is my first Visual Basic project. I’ve read many books while developing this program over the past few weeks. The most useful is certainly Paul Lomax’s *VB & VBA In A Nutshell* (O’Reilly, 1998, ISBN 1-56592-358-8).

Tutorials found at <<http://word.mvps.org/>> also proved helpful.

# Contact:

You can contact me using the information listed at the beginning and at the very end of this document.

I’ve developed GreekTranscoder on my own computer, using personal copies of the programs and books I needed to do so. Which turns this endeavor into an expensive and time-consuming hobby.

So, if this program proves useful to you, please consider making a [donation](http://www.greektranscoder.org/#donate) to show your support and appreciation and help me improve it further.

Last, if you need an AZERTY polytonic Greek Unicode keyboard layout for use with Mac OS X 10.2 or later, I’ve created one which is also available for download from GreekTranscoder’s [home](http://www.greektranscoder.org) as well as the École Normale Supérieure’s [web site](http://www.antiquite.ens.fr/txt/cea-ressources.htm#rubrique09). It’s free, just like GreekTranscoder.

Thank you for using GreekTranscoder!

David-Artur Daix  
Centre d’Études Anciennes  
Département des Sciences de l’Antiquité  
École Normale Supérieure  
45 rue d’Ulm, 75230 Paris Cedex 05

GreekTranscoder is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

GreekTranscoder is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You'll find a copy of the GNU General Public License along with GreekTranscoder.

1. I’ve given the name “LaserGreek” to the current Linguist's Software encoding. [↑](#footnote-ref-1)
2. When transcoding into Sgreek, the text produced is Betacode-compatible and does not use composed non-spacing diacritics. [↑](#footnote-ref-2)
3. I have given the name “SuperGreek” to the old (up to mid-1990s) Linguist's Software encoding. See the Technical notes section for details. [↑](#footnote-ref-3)
4. That is why Ralph Hancock’s Antioch will only convert documents using the Word 6/Word 95 format and not a later one. If your documents use such « symbol » fonts, you should register Antioch and use it to transcode them to Unicode. [↑](#footnote-ref-4)