בְּשָׁמַיִם

Ezra SIL Hebrew Unicode Fonts

Conversion Guidelines

SIL International
April 3, 2007

Table of Contents

Conversion Guide: Ezra SIL Hebrew Unicode Fonts

Introduction — About Encodings

The computer was designed to work with the English alphabet. Fonts originally had 128 slots for letters. In order to type data in a language other than English, the ABCs were often replaced with other letters. The result was called an encoding – assigning certain shapes to the 128 slots that are available in a font. Some encodings were standardized, such as ASCII and later ANSI, which allowed 256 slots in a font.

There are many different ways to encode Hebrew text on the computer. If data is typed with the SIL Ezra font, it is in a certain encoding. If it is typed with another Hebrew font, such as a commercial font, it will be in a different encoding. If you have an electronic copy of the Old Testament, it is likely in still another encoding.

Unicode seeks to provide one standard encoding with separate blocks for each writing system, such as Hebrew. A certain set of numbered slots have been set aside for specific Hebrew characters and marks. The Ezra SIL fonts follow the new Unicode encoding. This document will help you convert some of your old Hebrew data to the new Unicode numbers, so that you can use the Ezra SIL fonts without re-typing your data. It should be particularly useful to those who have made a significant investment in their data using the SIL Ezra fonts.

In this guide, we will explain how to convert from specific common encodings to Unicode. You can find instructions for installing programs that are mentioned here in the Installation Guide of the Ezra SIL release or on the NRSI website. While it is not necessary to be a programmer to follow these instructions, it is helpful to have some skill in that area if you need to adapt them for your particular situation.

About the Programs

SIL Converters and the Data Conversion macro

The SIL Converters package provides a means to select and use a converter (TECkit, CC and others) system-wide. However the only two parts of the package concern us.

1. The first is a Word macro, which provides a simple interface, making it easy to convert any file (e.g. SFM texts, lexicons, and even formatted Word documents) to a different encoding based on one or more TECkit maps or CC tables. It is in the form of a Microsoft Word document template. Attaching or adding it will affect the functioning of Word, including adding a menu item to the standard Tools menu. It may also initiate security questions regarding enabling or disabling macros. It should be treated like any other template or macro in this regard.
2. The second is the SFM File Converter, which is designed to convert SFM files. It is a stand-alone program with its own user interface.

The package also contains an editor and compiler for producing mapping files, should you wish to write your own.

TECkit

The TECkit package contains several implementations of the TECkit engine, including the DropTEC program. This program provides the interface for converting plain text files to Unicode and vice versa, via a mapping file. The TECkit package also contains an editor and compiler for producing mapping files, should you wish to write your own. This program does not make any changes to your operating system when installed and will not affect other programs.

Consistent Changes or CC

The Consistent Changes (CC) program is useful for finding all occurrences of specified characters, words, or phrases in a text file or series of text files, and making some type of change to this data in a consistent way. It was designed to work with plain text ASCII files, but can do limited conversion to Unicode.
This program does not make any changes to your operating system when installed and will not affect other programs.

About the Mappings

Three TECkit mappings are provided with the Ezra SIL release

- `SILEzratoUni50.map` and `SILEzratoUni50.tec`. This is used to convert Hebrew in SIL Ezra standard or display encoding into Unicode. If the Hebrew text is in 'display order' so that it displays correctly, then it will need to be reversed first (see below).
- `EzraSIL20to25.map` and `EzraSIL20to25.tec`. This is used to convert Hebrew in the 'old' Unicode encoding to the new standard. This is not essential, since text in the old Unicode will display acceptably with the new font, but some features will be less than optimal and searching may not give the expected behavior.
- `Hebrew_MCtoUni50.map` and `Hebrew_MCtoUni50.tec`. This is used to convert Hebrew in the Michigan-Claremont encoding used for WLC, OTA and CCAT texts into Unicode directly, without prior conversion to the SIL Ezra legacy encodings. Reversal should not be needed.
- The font package also contains several change tables for CC. These are updates of those distributed with the SIL Ezra legacy font.

Types of Conversion

Conversion of Plain Text Documents to Unicode

Plain text means your file contains only Hebrew, with no other language or information except possibly chapter and verse numbers. These types of files commonly have the extension "`.txt`." See the instructions for each type of conversion.

Type 1: Michigan-Claremont Plain Text to Unicode
Type 2: Plain Text Right-to-Left SIL Ezra Display Encoding to Unicode
Type 3: Other - Pointed Plain Text to Unpointed Plain Text

Conversion of Unicode Documents back to Plain Text

Type 4: Unicode to SIL Ezra Standard Encoding Plain Text – the return trip

Other Conversion

Type 5: Earlier version of *Ezra SIL* Unicode to *Ezra SIL* v2.5

Conversion of Other Documents to Unicode

Mark-up means that your data contains certain codes which indicate what type of data follows. A common mark-up is SFM (Standard Format Markers) which is in wide use by SIL. These types of files also are commonly saved with the extension "`.txt`." Another type of mark-up is RTF or HTML. Word has its own format when you save a file as "`.doc`". Some of these are addressed below.

Type 6: SFM Mark-up Text with SIL Ezra Standard or Display Encoding to Unicode
Type 7: SIL Ezra Standard or Display Encoding to Unicode and XML
Type 8: SIL Ezra Encodings in a Microsoft Word document to Unicode
Type 9: Earlier version of *Ezra SIL* Unicode in a Microsoft Word document to *Ezra SIL* v2.5

Installation and Setup in Microsoft® Windows®

If you have not done so already, follow the instructions in the Installation Guide to install the *Ezra SIL* fonts. It is not necessary, but may be helpful, to have also installed a Hebrew keyboard (a program for typing in Hebrew).
There are four programs used for conversion: Consistent Changes (CC), DropTEC, the SFM File Converter, and the Data Conversion macro. The URLs for downloading this free software are in the *Installation Guide* or listed below. You may not need all four, so you may wish to read through these instructions before performing all installations.

REMINDER: Uniscribe Update Required

To view fully-pointed Hebrew (with accents), you will need the version of Uniscribe (`usp10.dll`) that is at least as recent as the one released with Office 2003 (1.468.4015.0 or above). Having an updated Uniscribe will greatly reduce the number of dotted circles (U+25CC) which currently appear in Hebrew data. It will also improve the display of Hebrew diacritics in many cases.

Converting a Sample Text from the Westminster Leningrad Codex (WLC)

Type 1: Michigan-Claremont Plain Text to Unicode

The Westminster Hebrew Institute (WHI) http://whi.wts.edu/ "maintains the canonical version of the electronic representation of the best complete manuscript of the Hebrew Bible" and makes it freely available to anyone to ensure the integrity of the electronic texts. We recommend downloading this from WHI rather than using older and less reliable electronic texts.

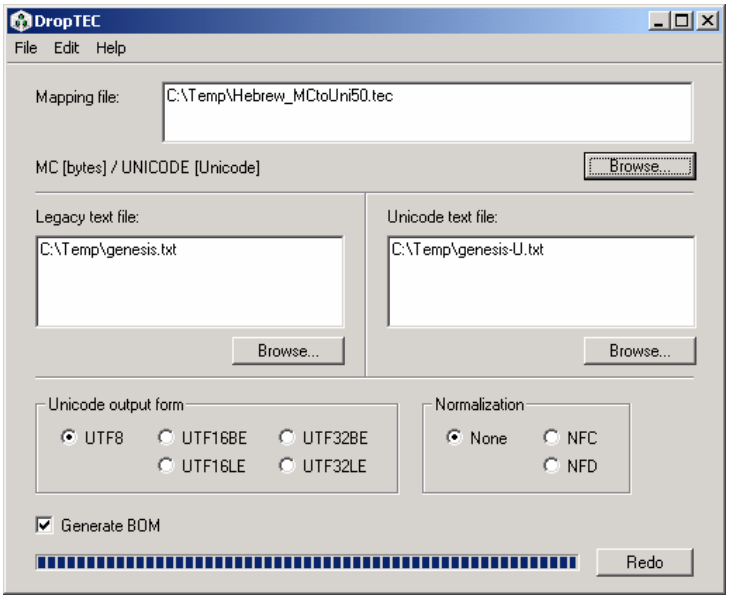
Getting and Converting the WLC Text

Please read through all the directions before beginning.

You can download a copy of the WLC in Michigan-Claremont encoding from the Westminster Hebrew Institute at the following website:
1. Go the website ftp://whi.wts.edu/ and navigate to ftp://whi.wts.edu/WestminsterLeningradCodex/
2. Download WLCmichigan.zip and decompress the file using the "zip" function in Windows XP, or with WinZip® (PC) or Stuffit Expander® (Mac) to decompress the file.
3. Other formats are available on the same web-site, but as of March 2, 2007, they claim that the most reliable format is the version in Michigan-Claremont encoding.

Conversion of the WLC text

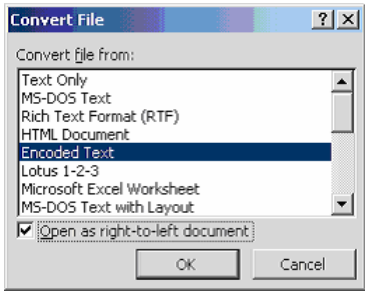
Once you have successfully downloaded the WLC text from WHI, you can then proceed with the conversion.
1. Open the folder where you have stored the data files. and rename one of the WLC files, e.g. `gn.wlc46.txt` to `genesis.txt`. In this example, we are using `C:\TEMP` as the folder location.
2. Next convert the file to Unicode using the TECkit program `DropTEC.exe`. Note that you *must* leave Normalization as **None**, and **Generate BOM** must have a checkmark. **UTF-8** is the correct selection for Microsoft Office 2000, 2002, and 2003. Other operating systems and applications may use other formats.
3. The TECkit program supports "Drag-and-Drop" but you must *first* drag the mapping file (`Hebrew_MCtoUni50.tec`), and *then* the input file (`genesis.txt`) to the window. It will ask for an output file name and then run the conversion.



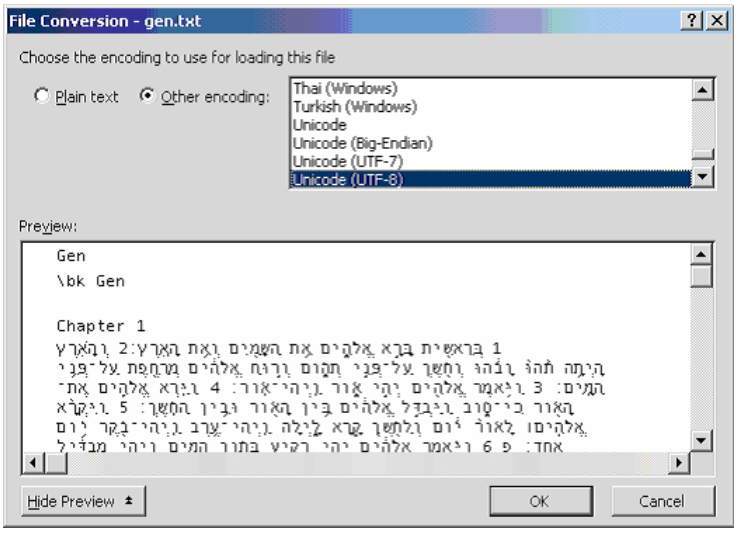
4. Open the file `genesis-U.txt` to see the results. Directions are given below for Opening a File in Word 2000 and Word XP, since Word now offers more options when it opens a Unicode file. Your Unicode file can be renamed and saved as other file types, as you desire.
This is the end of instructions for *Type 1: Michigan-Claremont (Plain Text) to Unicode* conversion.

Opening a Unicode text File - Word 2000

After you have run through the plain text conversion directions, open `genesis-U.txt` in Word. At the first box, make sure “Encoded Text” is highlighted. Check the “Open as right-to-left document” box at the bottom. Click **OK**.

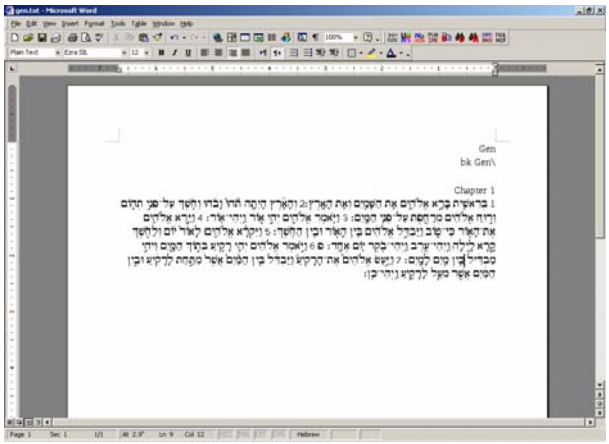


In the next window, **Other encoding** should be selected and be sure “Unicode (UTF-8)” is highlighted. Click **OK**.



Edit / Select All and right-align the text using the **Paragraph** button that points left (Right-to-Left).

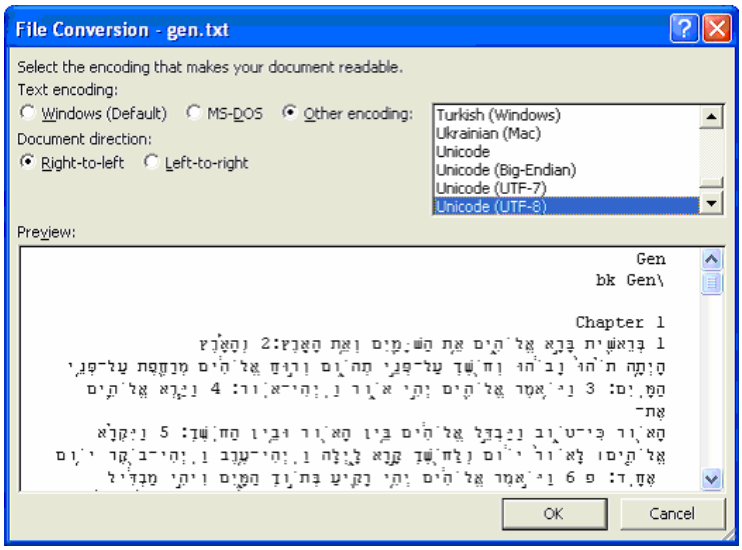
With the text still selected, change the font to *Ezra SIL* and choose a viewable point size. With all text still selected, double-click the box that says what language this is and if necessary, change to **Hebrew**. It is located at the bottom center of the Word screen on the same line as **Page** and **Sec**. It may say **Arabic Saudi Arabia**. Here is a shorter text as an example:



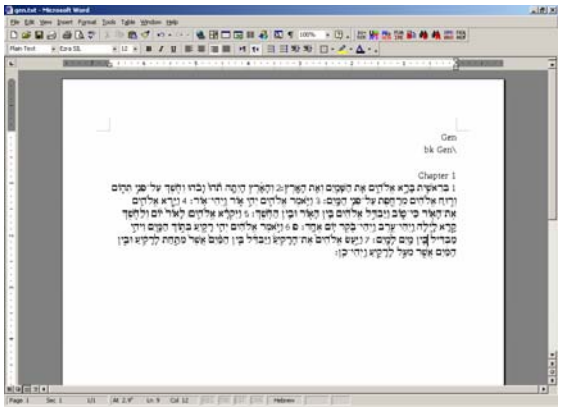
If the cursor is in an English section, it will say **English**. Click on a Hebrew text line to check that the language name changes to **Hebrew**.

Opening a Unicode text File - Word XP

Open `genesis-U.txt` in Word 2002. Make sure “Other encoding” is selected and “Unicode (UTF-8)” is highlighted. Document direction should be “Right-to-left.” Click **OK**.



Edit / Select All, change the font to *Ezra SIL* and choose a viewable point size.
With all text still selected, double-click the box that says what language this is and if necessary, change to **Hebrew**. It is located at the bottom center of the Word screen on the same line as **Page** and **Sec**. It may say **Arabic Saudi Arabia**. Below is a shorter text as an example:

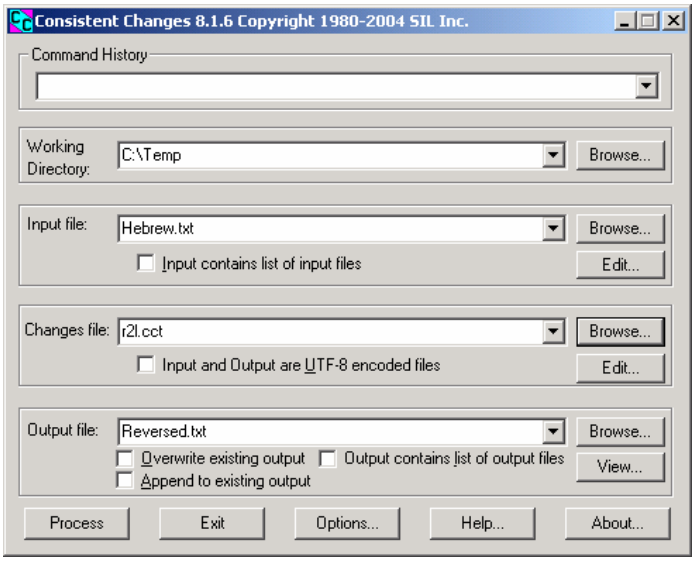


If the cursor is in an English section, it will say **English**. Click on a Hebrew text line to check that the language name changes to **Hebrew**.

Type 2: Plain Text Right-to-Left SIL Ezra Display Encoding to Unicode

If you have your own data that is in right-to-left order, in order to display correctly on the screen with the old SIL Ezra fonts, you must first reverse the text.

1. Start by converting the line direction with the CC program `r2l.cct`. Fill in the CC window as shown below and click on **Process**. Note that this program does not handle drag-and-drop.



Note that the `r2l.cct` program will reverse every line in the file, regardless of its language or directionality.

2. Then use the TECkit *DropTEC* to convert to Unicode, as described in the Type 1 instructions, but using `SILEzratoUni50.tec`. Unicode applications, such as Word 2003, which support right-to-left languages, will display it in the correct order on the screen.

This is the end of instructions for *Type 2: Plain Text Right-to-Left SIL Ezra Display Encoding to Unicode* conversion.

Type 3: Other — Pointed Plain Text to Unpointed Plain Text

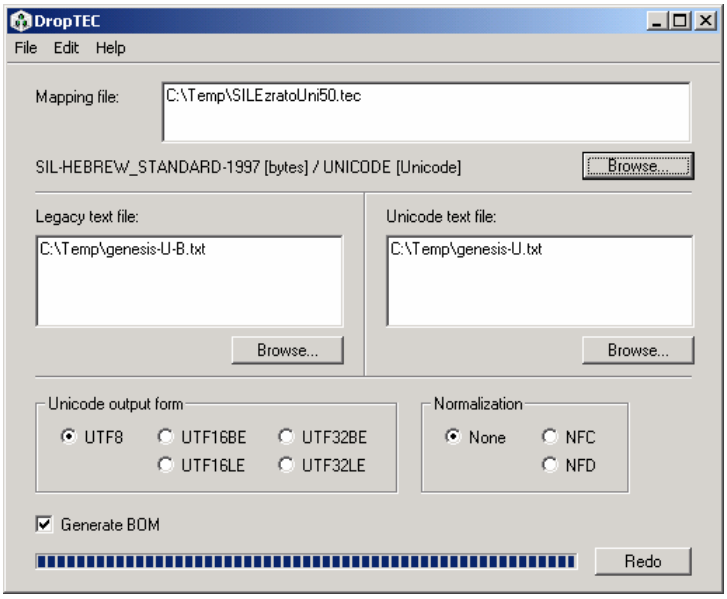
If you wish to have unpointed text (no vowels or cantillation), remove the pointing from a standard encoding file using the CC program `unpoint.cct`. Reverse the text, if necessary, as in the Type 2 conversion, then convert it to Unicode, as described in Type 1 conversion above.
This is the end of instructions for *Type 3: Pointed Plain Text to Unpointed Plain Text* conversion.

Type 4: Unicode to SIL Ezra Standard Encoding - the Return Trip

The TECkit mapping can also convert Unicode Hebrew data back to SIL Ezra SE, with certain qualifications:

- Accents that were in high-low order will now be in low-high order.
- *Meteg* that was originally coded as right meteg will be regular *meteg* when it occurs with *holem* or *shureq*.
- Anything that was ambiguous in Unicode cannot be fixed.
- Data converted from WLC to Unicode and back will have spaces around the verse numbers.
- If the data was incorrectly encoded in the original, the return trip may correct the error. One example is *shureq* with a vowel. This becomes *vav* + *dagesh* with a vowel where it can be determined from the context.
- There were some errors in the original `bhs2se.cct` table which have been corrected in this release. If possible, use `MC2se_2007.cct` to re-convert your WLC text to SIL Ezra standard encoding.

To do the return trip from Unicode to `SIL`Ezra standard encoding using `DropTEC.exe`, simply drag the mapping file `SILEzratoUni50.tec` into the top box and the Unicode text file `genesis-U.`txt to the right-hand box “Unicode text file.” The conversion from Unicode to Standard Encoding will ask for an output file name and then begin.



Then convert the standard encoding to display encoding using `se2de_2007.cct` and CC and reverse the text to give display order.
This is the end of instructions for *Type 4: Unicode to SIL Ezra Standard Encoding* conversion.

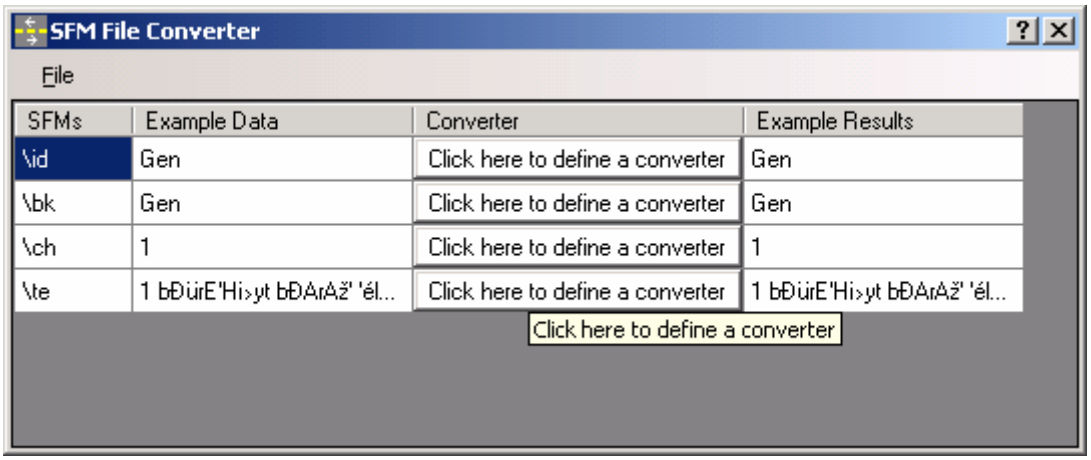
Type 5: Earlier versions of Ezra SIL Unicode to Ezra SIL v2.5

If you have data typed in version 1 or version 2.0 of the *Ezra SIL* Unicode font, you may wish to update your texts to v 2.5. This is not essential; text typed in version 2.0 will still be readable using version 2.5 of the font. A stricter data order and use of control characters is required in the later versions, so the display will be better if the text is updated. See the *Keying in Hebrew* document found in the Documentation folder. Certain data combinations will look incorrect in version 2.5 of the font, unless the order is changed to meet the new requirements.
We have provided a TECkit mapping `EzraSIL20to25.tec` for updating from earlier versions to version 2.5 of the *Ezra SIL* fonts.

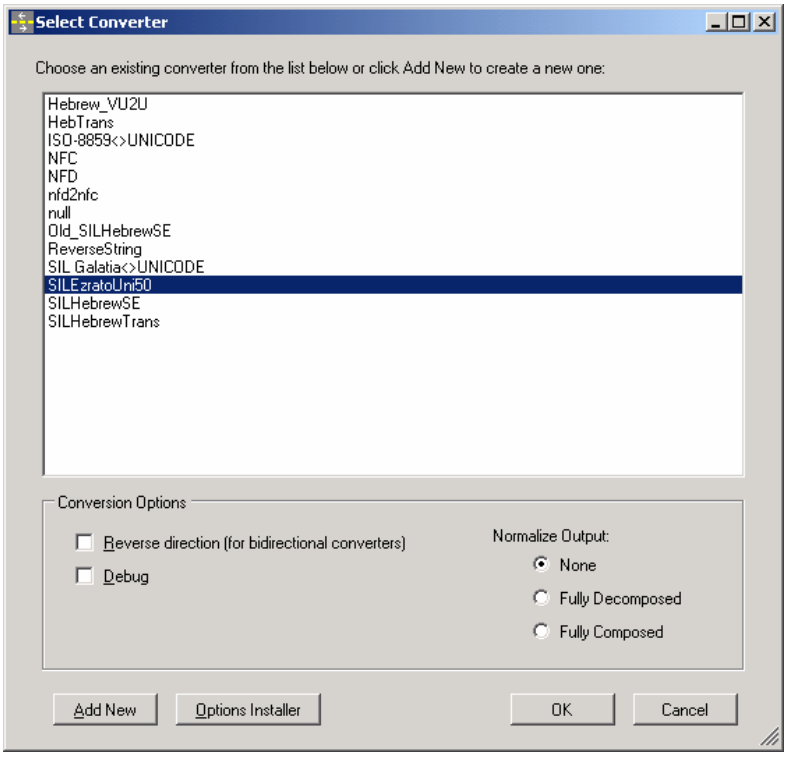
Documents with Formatting:

Type 6: SFM Mark-up Text with SIL Ezra Standard or Display Encoding to Unicode

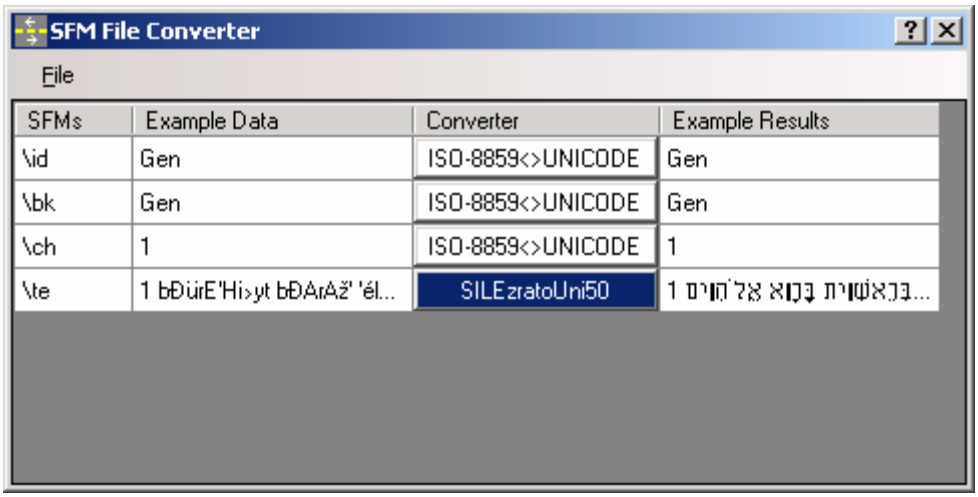
SFMs are Standard Format Markers. These are used extensively in SIL to indicate what type of data follows. For example, the SFM "\v" marker might indicate the verse number follows. If you have a text file containing SFMs with *SIL Ezra* data in some fields, you can convert them to Unicode using the SFM File Converter.
1. In SFM File Converter, open your SFM file as a Non-Unicode document.



2. For each field, click opposite it in the 'Converter' column and select the required converter.



3. If the converters you require are not already installed, install them following the instructions with the Converters package.
4. For Hebrew fields, use `SILEzratoUni50`.
5. Even non-Hebrew fields must be converted. By default this should be `ISO-8859<>UNICODE`



6. To perform the conversion use the File Menu – Convert and Save, and select UTF-8. The conversion will be carried out and you will asked for a filename for the output.
This is the end of instructions for *Type 6: SFM Mark-up Text with SIL Ezra Standard or Display Encoding to Unicode* conversion.

In this package we have also provided a sample control file `HEB-map.xml` as a starting point for creating a control file to meet your specific needs for working with Hebrew and the command-line program **Sfconv.exe**. However, the SFM File Converter has its own user interface and is simpler to use.
For more information, there is a tutorial on "Structured Data Conversion" on the SIL website: http://scripts.sil.org.

Type 7: SIL Ezra Standard or Display Encoding to Unicode and XML

We are still in the research stage of working with XML. However, you may be interested in the tutorial "An Experiment in Converting Legacy Data to Unicode and XML" found on the SIL website: http://scripts.sil.org.

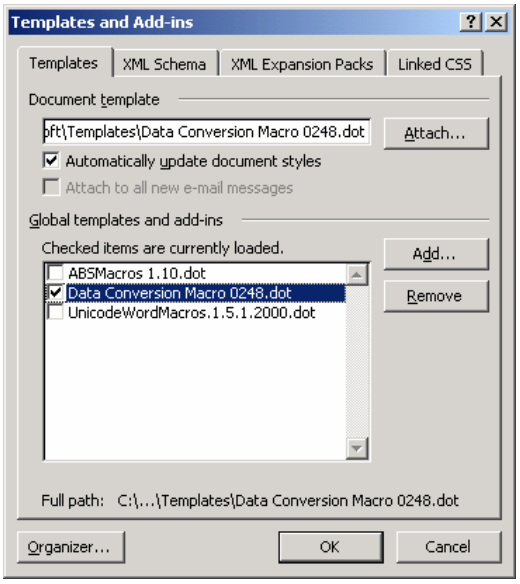
Type 8: SIL Ezra Encodings in a Microsoft Word document to Unicode

One part of SIL Converters 2.5 <http://scripts.sil.org/EncCnvtrs> is a Visual Basic macro that runs in Word, called "Data Conversion". It allows the user to convert the whole of a Word document or only text in a specified font or in a specified style. Once attached, the macro appears as "Data Conversion" on the Tools menu. The Data Conversion macro requires only a moderate level of computer expertise to install and use, although the documentation may appear to be more technical.

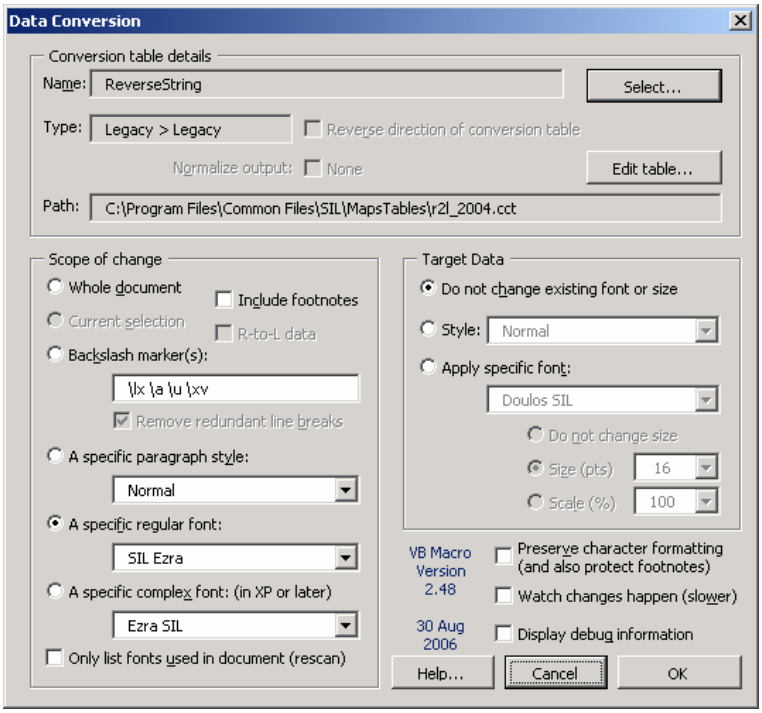
Always work on a copy of your data file. There is no "Undo".

1. First, add or attach the Word template to your document. In Word, go to Tools Menu – Templates and Add-ins. If `Data Conversion Macro 0248.dot` is not displayed, click on the

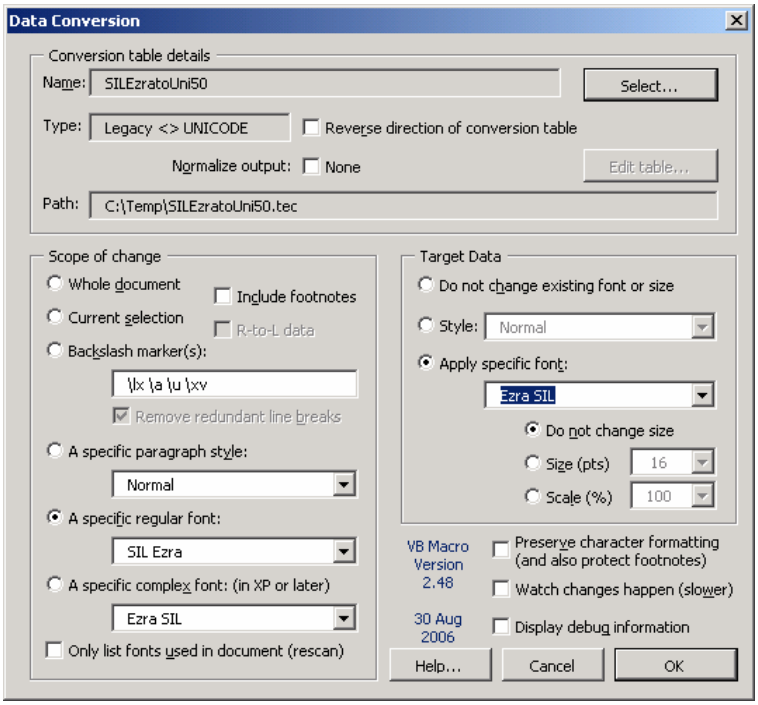
'Add' button, and navigate to find the `.dot` file. Ensure that the check box next to it is checked and click on OK. There should now be an item on the Tools Menu `Data Conversion`.



2. It is likely that the SIL Ezra encoded Hebrew is in display order in order for it to be displayed in Word correctly as right-to-left text. Before it is converted to Unicode, it must be reversed in direction. In the Tools Menu, select Menu `Data Conversion`. Complete the window as below, using the Select button to select the `ReverseString` converter, and the radio buttons to apply it only to text in the SIL Ezra font, and with no change of font.



3. As a second step, the Hebrew can be converted to Unicode using the `SILEzratoUni50` converter, applied only to text in the SIL Ezra font, and with change of font to *Ezra SIL*.



This is the end of instructions for *Type 8: Microsoft Word documents to Unicode* conversion.

The conversion of data from SIL Ezra SE to Unicode appears to work well. The conversion back (place a checkmark by "Reverse direction of conversion table") does work, but Word may have trouble properly displaying the results. This is because it may be holding over the directionality (RTL) from the original Unicode font. An example is the *petuha* character d62 which incorrectly displays as ">". If this occurs, the text needs to be marked as LTR. The "Set Run Ltr" macro available in "ABSMacros" will correct this. This macro is available on the http://scripts.sil.org/cms/scripts/page.php?site_id=nrsi&item_id=RTL_in_MSOffice site. Note that the Data Conversion "reverse" is not referring to right-to-left line direction, but rather to a conversion from Unicode back to legacy (old) data. Once you have the data converted back to Ezra SIL SE (Standard Encoding), you will need to use the ReverseString converter to reverse the line direction, if you wish to display the text.

See also the tutorial "Structured Data Conversion" on the SIL website: http://scripts.sil.org.

Type 9: Earlier versions of Ezra SIL Unicode in a Microsoft Word document to Ezra SIL v2.5

This uses the same Data Conversion macro as the Type 8 conversion, but with `EzraSIL20to25` as the converter applied only to text in the Ezra SIL font and without change of font.

On Canonical Combining Classes

Numeric classes are assigned in Unicode to each character for a language. These classes, called canonical combining classes, were originally meant to assist in sorting—to establish without question, whether two words (or strings of data) were equivalent. This is especially pertinent to a language which uses accents, (unlike English). For example, canonical ordering would be used to determine whether "a" + "`" was the same as "à." *Sorting* order is not the same as *store* order (the order the characters are physically stored in a file). For example, "càt" could be stored "c", "a", "`" "t", or "c", "`", "a", "t", or "c" "à" "t"). While it was not originally the intended use of canonical classes, the World Wide Web Consortium is talking of requiring the store order be the same as the sorting (canonical) order. Since the canonical ordering for Hebrew bears little resemblance to any store order now in use, and we, the font developers, found it impossible to code, we have used a different store order. See the document "`Keying in Hebrew.pdf`" for information on how characters should be stored for correct display with the Ezra SIL fonts. The TECkit mapping provided in this release sorts SIL Ezra standard encoding data into the correct order for the Ezra SIL v?? fonts when it does the conversion to Unicode. Hebrew data in canonical order or in normalization form C or D will not display correctly with Ezra SIL fonts. See http://www.unicode.org for more information about canonical orders and normalization.

With Ezra SIL v.2.5, the expected store order is more restricted than with v.1, but is compatible now with a number of other fonts, including *Vusillus* (by Ralph Hancock), *SBL Hebrew* (Society of Biblical Literature and Tiro Typeworks), and eventually Microsoft.

Technical Support

As these programs are provided free, we cannot offer a commercial level of support. However, if you find errors or other problems using the Ezra SIL Hebrew Unicode Fonts, we would like to know. We can be contacted at:

User Support
SIL International Publishing Services
7500 W. Camp Wisdom Rd.
Dallas, TX 75236
USA
E-mail: sil_fonts@sil.org