בַּשָּׁמַיִם

Ezra SIL Hebrew Unicode Fonts

Conversion Guidelines

SIL International

January 29, 2004

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

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.

Data Conversion and Encoding Converters

This package provides a means to select and use a converter (TECkit, CC, or ICU-based) system-wide. However the part of the package we are using is a Word macro, which provides a simple interface, making it easy to convert any file (e.g. SFM texts, lexicons, and even Word documents) to a different encoding based on one or more TECkit maps or CC tables.

This program is in the form of a Microsoft Word document template. Installing 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.

SFconv

SFConv is a command-line utility which can convert sfm files to and from Unicode.

This program does not make any changes to your operating system when installed and will not affect other programs.

TECkit

The TECkit package contains the DropTEC program. This program provides the interface for converting plain text files to Unicode and vice versa, via a mapping file. The Hebrew mapping is provided with the Ezra SIL release. 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.

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 SIL Ezra Standard Encoding 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

Other Conversion

Type 5: *Ezra SIL* v1.0 Unicode to *Ezra SIL* v2.0 Unicode

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: Microsoft Word documents to Unicode

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), TECkit, SFconv, 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 was released with Office 2003. 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 Biblia Hebraica Stuttgartensia (BHS)

Type 1: Michigan-Claremont (CCAT) Plain Text to SIL Ezra Standard Encoding to Unicode

There are several formats of electronic texts available either free or for purchase on the Internet. This example uses the one from the Oxford Text Archive (OTA). It is an older copy of the Biblia Hebraica Stuttgartensia but is still quite useful. SIL does not provide copies of the biblical texts to the public since they are copyrighted, licensed and available from other sources.

To use these instructions for converting files already in Ezra SIL standard or display encoding, start at Step 3 under "Conversion of the BHS Text" and substitute your filename for the input file.

Getting and Converting the BHS Text

Please read through all the directions before beginning.

You can download a copy of the BHS in Michigan-Claremont encoding from the Oxford Text Archive at the following website:

http://ota.ahds.ac.uk/

1. Go the website above.

2. Type the word "Biblia" in the Quick Search box and click FIND.

Look for the item which says:

Biblia Hebraica Stuttgartensia : (Michigan-Claremont text)

3. Click to put a checkmark by the title. Click "Download Selected Texts." Agree to the terms and conditions by reading them, checkmarking the box, and giving your email address.

4. Under the banner "GZip'ed Tar File," click "Download." **Save** the file.

5. Use WinZip® (PC) or Stuffit Expander® (Mac) to decompress the file. The filename for the text is "`biblheb.525`."

Create a Sample Text

You should select a section of text you wish to work with, for example, the book of Genesis, and copy it to another working file, such as `genesis.ota`. To do this:

1. Open `OTA.BHS` in Word or a text editor capable of handling large files. This may take a few minutes.

2. Search for "Exo."

3. Click once just above this line, at the end of the Genesis text.

4. Without clicking again on the text, scroll to the top of the file.

5. **Shift-click** once before the first character of text. This should highlight the entire text of Genesis. If not, try again.

6. Press **Ctrl-c** to copy, **Open** a new document, and **Ctrl-v** to paste.

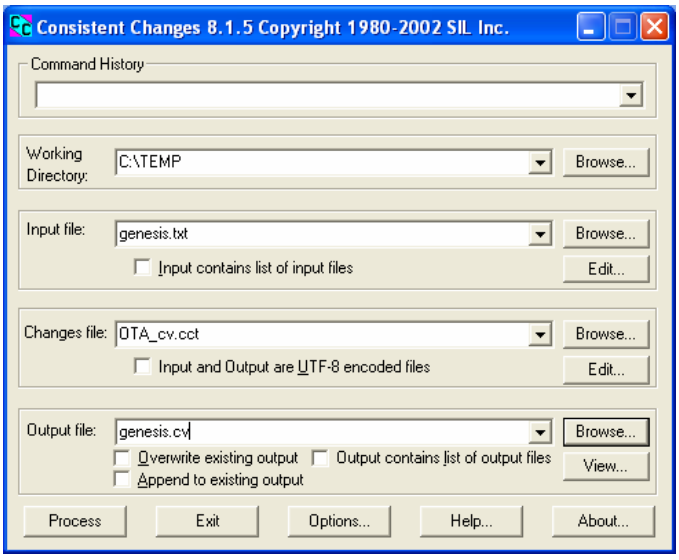
7. **Save** this document as `genesis.txt`, as a Plain Text (*.TXT) file (NOT a `.doc`!). Choose **Windows (default) — Western European (Windows)** as the encoding, if it asks. You are telling Word to save the file as a regular plain text file. You don't wish to have it converted to another language or computer encoding, something that is now an option in Word.

Conversion of the BHS text

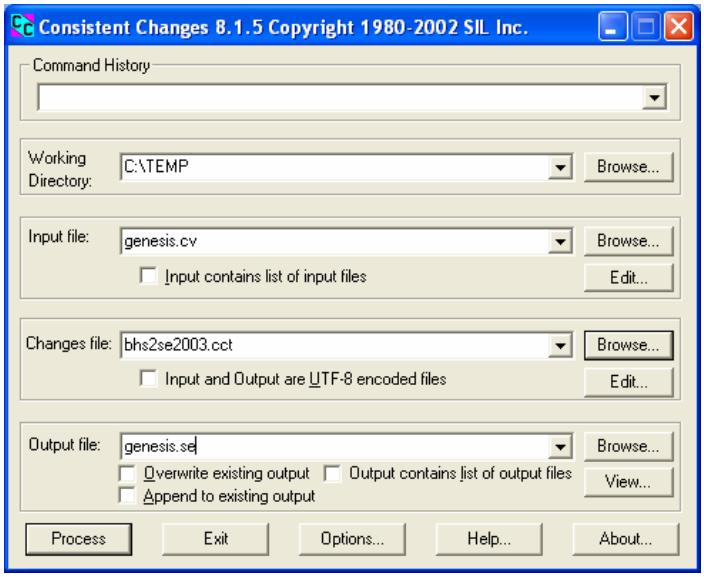
Once you have successfully downloaded the BHS text from the Oxford Text Archive, you can then proceed with the conversion. If you already have a text in Michigan-Claremont encoding *with verse numbers*, you can skip step 2.

1. Open the folder where you have stored the data file `genesis.txt`. In this example, we are using `C:\TEMP` as the folder location.

2. First, add the verse numbers (cv stands for "chapter-verse") by double-clicking the Consistent Changes program `CCW32.exe` or its icon. Fill in the window as shown below and press **Process**. Note: this program does not handle "Drag-and-Drop."

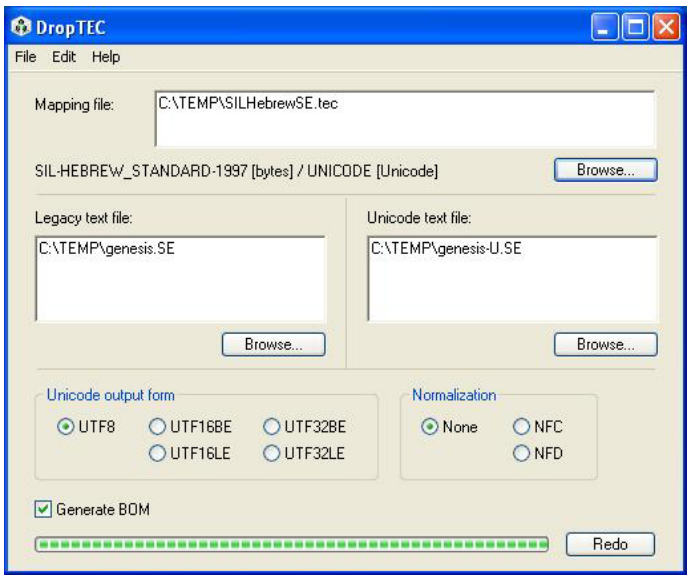


3. The next step converts the OTA text to SIL Ezra SE (standard encoding). Fill in as shown below, and click **Process**:



4. 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.

5. The TECkit program supports "Drag-and-Drop" but you must *first* drag the mapping file (`SILHebrewSE.tec`), and *then* the input file to the window. It will immediately run the conversion.

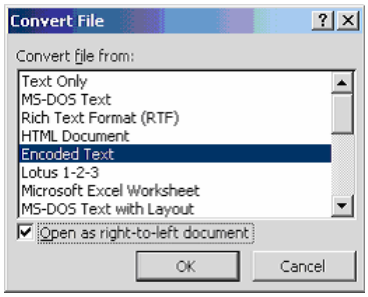


6. Open the file `genesis-U.se` to see the results. Directions are given below for Opening a File in Word 2000 and Word XP, since Word now offers more options when it opens a Unicode file. Your Unicode file can be renamed and saved as other file types, as you desire.

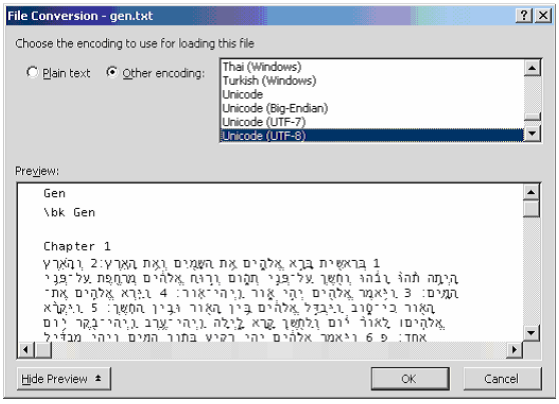
This is the end of instructions for *Type 1: Michigan-Claremont (Plain Text) to SIL Ezra Standard Encoding to Unicode* conversion.

Opening a File - Word 2000

After you have run through the Type 1 or Type 2 plain text conversion directions, open `genesis-U.se` 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**.



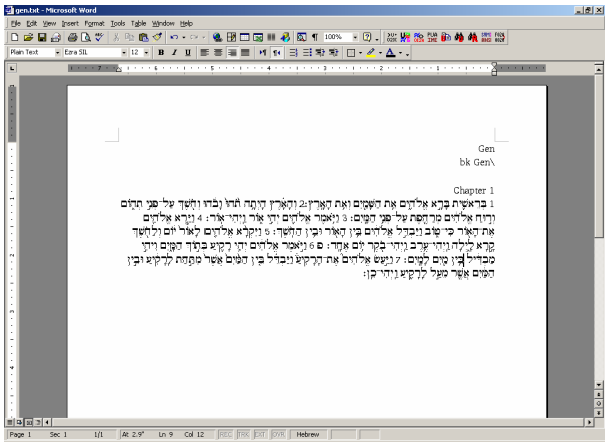
In the next window, **Other encoding** should be selected and be sure "Unicode (UTF-8)" is highlighted. Click **OK**.



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.

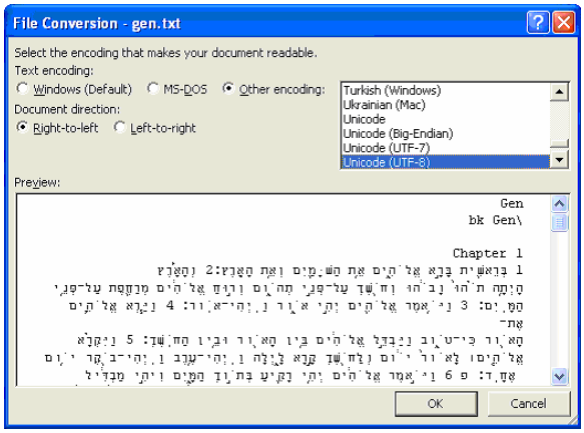
With all text still selected, double-click the box that says what language this is and if necessary, change to **Hebrew**. It is located at the bottom center of the Word screen on the same line as **Page** and **Sec**. It may say **Arabic Saudi Arabia**. Here is a shorter text as an example:



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**.

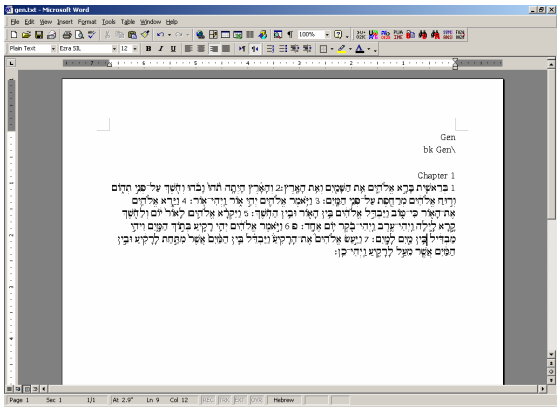
Opening a File - Word XP

Open `genesis-u.se` 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 **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 reverse the text. Start by converting the line direction with the CC program `r2l.cct`. Use the CC example in the Type 1 conversion above as an example. Note that the `r2l.cct` program will reverse every line in the file, regardless of its language or directionality.

Then do the TECkit *DropTEC* conversion to Unicode, as described in the Type 1 instructions. 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 your standard encoding file using the CC program `unpoint.cct`. 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.

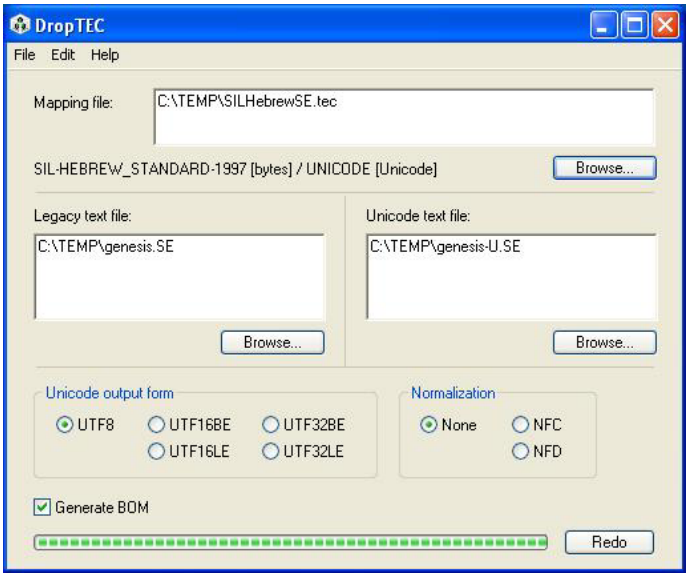
Type 4: Unicode to SIL Ezra Standard Encoding - the Return Trip

The TECkit mapping can also convert Unicode Hebrew data back to Ezra SIL 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. *Left pashta* (d143 or decimal 143) that was displayed medially will now be encoded as *medial pashta* (d137).

- Data converted from OTA 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. Use `bhs2se2003.cct` to re-convert your BHS text, where possible.

To do the return trip, simply drag the Unicode file `genesis-U.se` to the right-hand box "Unicode text file." The conversion from Unicode to Standard Encoding will begin immediately.



This is the end of instructions for *Type 4: Unicode to SIL Ezra Standard Encoding* conversion.

Type 5: Ezra SIL v1.0 Unicode to Ezra SIL v2.0 Unicode

If you have data typed in version 1 of the *Ezra SIL* Unicode font, you may wish to check your texts for characters which are different in v2. All Private Use Area (PUA) characters assigned by SIL in v1.0 have been removed from v2.0 of the font. Private Use characters will not function properly in most software currently available. Use the characters on the right side of the chart to replace those on the left.

Ezra SIL v.1	**Ezra SIL v.2**
U+F300 HEBREW REVERSED NUN	nun+CGJ+combining dot above (U+05E0 U+034F U+0307)
U+F301 HEBREW MARK LOWER DOT	combining dot below (U+0323)
U+F302 HEBREW ACCENT RIGHT METEG	meteg+CGJ or ZWNJ+vowel
U+F303 HEBREW ACCENT LEFT METEG	hataf vowel+CGJ or ZWNJ +meteg (hataf (U+034F or U+200C) U+05BD).

A stricter data order is required in version 2. See the *Keying in Hebrew* document found in the Documentation folder. Certain data combinations will look incorrect in version 2 of the font, until the order is changed to meet the new requirements.

We have not provided a TECkit mapping for version 1 to version 2 of the *Ezra SIL* fonts. A simple "search and replace" should handle this conversion, if needed.

Documents with Formatting:

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

In this package we are providing a sample control file `HEB-map.xml` for working with Hebrew and the **SFconv** program. 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 use the included control file as a starting point for creating a control file to meet your specific needs. **SFconv** is a simple program to use and if you are already familiar with the SFMs you use, it should not be difficult to work with.

If you have installed **SFconv**, copy `sfGen.txt` and `HEB-map.xml` to your working folder. Then open a command prompt window by clicking **Start / All Programs** (or **Programs**) **/ Accessories / Command Prompt**. Navigate to your working folder and type:

```
sfconv -8u -c HEB-map.xml -i sfGen.txt -o sfGen.out -bom
```

SFconv will create the output file `sfGen.out`. That's all there is to it.

Another method is to create a plain text file with the command above, and save it with a "`.bat`" extension. Do not use "`sfconv.bat`" as the filename. Double-click the `.bat` file. It will open the **Command Prompt** window, run the command line, and close. Use **Ctrl-C** to stop the program if it gets in a loop.

Another alternative, if you have a **Command Prompt** window already open is to run the `.bat` file at the prompt. Just type the filename, without the "`.bat`" extension, and press **Enter**. In that case, the window will stay open with the prompt ready for the next command.

You can also place the **sfconv** program with your other program files and put in the full path name in the command line or add it to your PATH.

See the "`TECkit version 2.doc.pdf`" which comes with the TECkit package for more information about using the control file and for working with in-line markers.

> **Tips**: Note that the character "`|`" is a valid *SIL Ezra* Hebrew character and should not be used as an in-line marker. A safer choice might be "`~`", provided your old *SIL Ezra* Hebrew data does not use the *masora* circle (d126).
>
> Don't forget to use the command line switch "-bom" when converting to Unicode. This makes sure there is a byte order marker included in the file.
>
> Always convert any Latin data to Unicode also. An example of how to do this is included in the control file `HEB-map.xml`.

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: Microsoft Word documents to Unicode

Also in development is software for simple Word conversions. See the tutorial “Structured Data Conversion” on the SIL website: http://scripts.sil.org. This software allows the user to convert the whole document or only text in a specified font or in a specified style.

The tutorial refers to a Visual Basic macro called “Data Conversion”. It is available on the website on the page titled “Microsoft Word/COM support for TECkit, CC, and ICU.” Search for “encoding converter.” The macro runs in Word and when installed, will appear as “Data Conversion” on the Tools menu. The Data Conversion macro requires a moderate level of computer expertise to install and use, although the documentation may appear to be more technical.

Tips: Name a folder to extract all the files from *EncCnvtrs.v###.zip*.

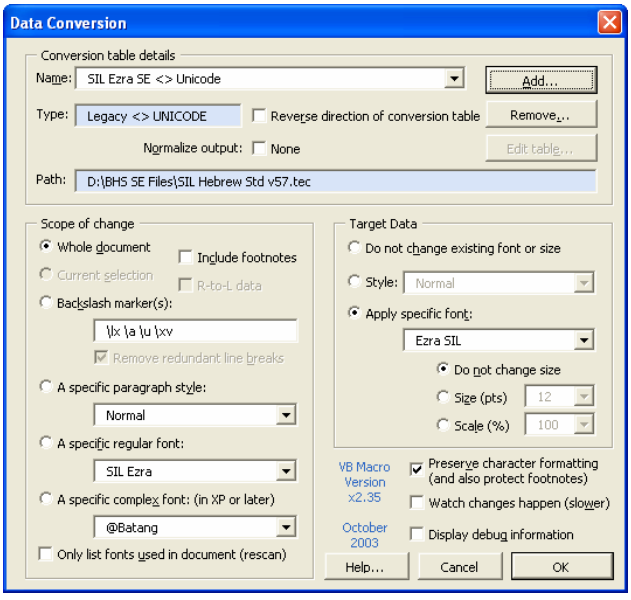
Run *setup.exe* to install the Encoding Converters program.

Copy the file *Data Conversion Macro x(###).dot* to **Program Files / Microsoft Office / Office 10 / Startup** if you are running Windows XP.

Copy the file *Data Conversion Macro x(###).dot* to **Program Files / Microsoft Office / Office / Startup** if you are running Windows 2000.

If **Tools / Macro / Security...** is set to Medium, *Word 2002* will ask whether to enable macros for this macro. You should click **Enable Macros**.

Here is an example of what the Data Conversion window looks like:



You must click **Add** to add the "SIL Ezra SE <> Unicode" conversion mapping to the list of encoding converters available. The mapping filename is *SILHebrewSE.tec*.

Note that you can convert the whole document or only data in a specific font.

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 may wish to use `r2l.cct` to reverse the line direction, if you wish to display the text.

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

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 v2.0 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, the expected store order is more restricted than with v1, 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

Phone: (972) 708-7495
FAX: (972) 708-7388
E-mail: sil_fonts@sil.org