Emdros Chunking Tool User's Guide

Ulrik Petersen

August 15, 2009

Contents

1 Introduction

This is the User's Guide to the Emdros Chunking Tool.

The Emdros Chunking Tool allows you to split running text into lines and then indent those lines with respect to each other.

1.1 Two versions

The Emdros Chunking Tool exists in two graphical versions:

- A non-Unicode version
- A Unicode-aware version

1.2 Getting started

1.2.1 Introduction

Before running the Emdros Chunking Tool for the first time on a database, you need to write a configuration file that matches your database. This is a one-off thing: Once it's done, you don't need to bother with it any more.

1.2.2 Sample configuration file

A number of sample configuration files are supplied with Emdros. You can use these as a starting point for writing your own configuration file.

One supplied configuration file is called "tisch.cfg", while another is called "wihebrew.cfg". You can search for these on your computer to locate where they are, or see the manual page for chunkingtool to know where they are installed (on Windows, they are installed in

1.2.3 Full details

The "tisch.cfg" file is almost self-documenting. However, you can get more information about the details of the configuration file here:

- Configuring the program

2 Using the program

We now explain how to use the program.

2.1 Starting the program

2.1.1 Getting started

Once you open the program, you will be presented with the main screen. You will then need to "connect" to a database. Either choose the menu-item "File|Connect" or press the button  "Connect to database".

2.1.2 Connection dialog

You will then be given a dialog box allowing you to choose the Connection Settings. At the top is a drop-down box allowing you to choose the backend. Based on this choice, the dialog box will appear slightly differently depending whether the backend is:

- SQLite 2 or 3, or
- MySQL or PostgreSQL

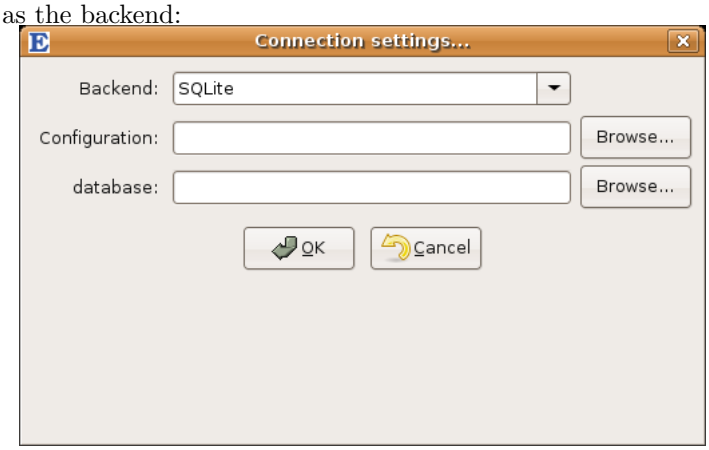
2.1.3 Non-Unicode vs. Unicode

For each backend, the program exists in two versions:

- A non-Unicode version
- A Unicode-aware version

2.1.4 SQLite version

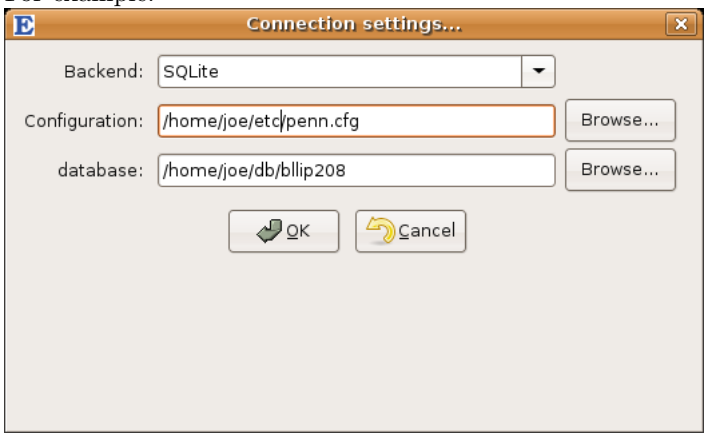
The Connection Settings dialog looks like this if you have selected SQLite 2 or 3 as the backend:



Set the configuration file The first thing you should do is press the "Browse" button next to the "Configuration" edit box, then navigate to where you have your configuration file.

Once you've opened the configuration file, the "database" field will be filled from the "database" value stored in the configuration file, if any. If this is not the database you want, simply enter (or browse for) the database you want.

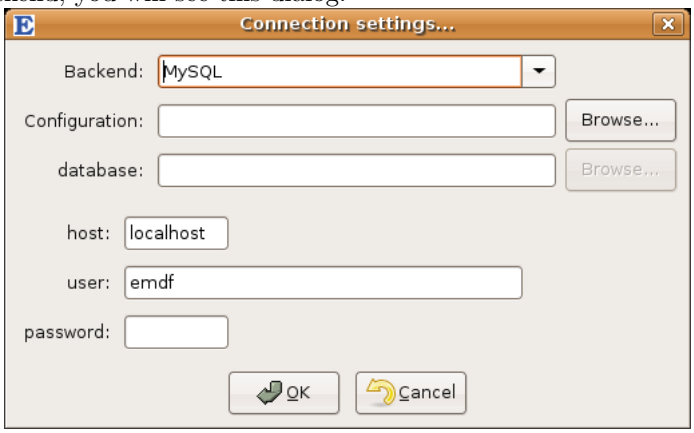
For example:



Press OK Once you're done setting the configuration file and the database, press "OK". If you want to quit the program instead, press "Cancel".

2.1.5 MySQL/PostgreSQL version

When you start the Emdros Chunking Tool using the MySQL or the PostgreSQL backend, you will see this dialog:



Set the configuration file The first thing you should do is press the "Browse" button next to the "Configuration" edit box, then navigate to where you have your configuration file.

Database Once you've opened the configuration file, the "database" field will be filled from the "database" value stored in the configuration file, if any. If this is not the database you want, simply enter the database you want.

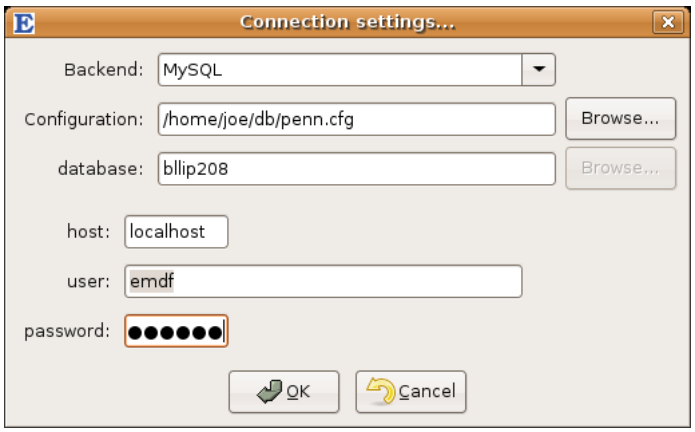
Host, user, password Most people can leave the "host" and "user" fields as they are, and simply write the password.

The "host" field shows which computer to connect to, i.e., the computer where the MySQL or PostgreSQL backend is running. "localhost" means the computer where the Chunking Tool is running.

The "user" field is the database user to connect to the backend as. Note that this may be different from your computer user name. The default is "emdf", since that is the recommended default user to create when you bootstrap the MySQL or PostgreSQL database (see "bootstrapping.txt" in the Emdros documentation).

The "password" field is for the password of the database user to connect as. This is set either by the database administrator, or by the one who bootstrapped the MySQL or PostgreSQL database.

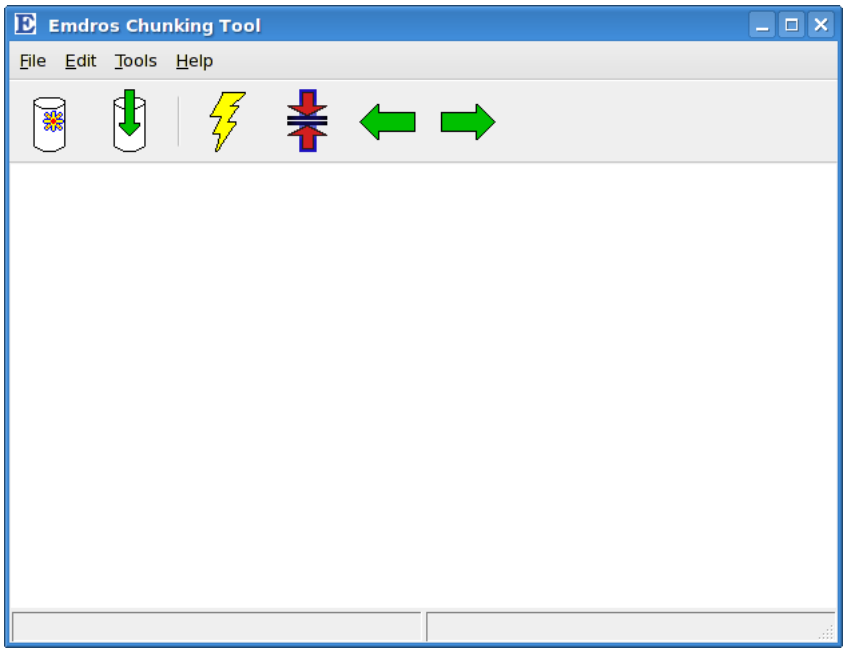
Example



Press OK Once you're done setting the configuration file and the database, press "OK". If you want to quit the program instead, press "Cancel".

2.2 The main screen

Once you've pressed "OK" on the "Connection settings" dialog box, you will see the main screen:



2.2.1 Parts

The main screen consist of these parts:

- A menu at the top.

  File Edit Tools Help

- Below that, a toolbar with buttons.

- Below that, an area that is to display the text you want to chunk. In the picture above, it is all white.

2.2.2 Next

Next, we describe each of these parts.

2.2.3 Toolbar

The toolbar looks like this:

The buttons represent actions, each of which will be explained below.

New Connection **New Connection**  : Connects to a database via a configuration file

Commit **Commit**  : Saves the chunking-changes made in the program, by committing them to the database.

Split, Combine **Split**  : Splits the current chunk right before the currently selected box.

Combine  : Combines the chunk containing the currently selected box with the previous chunk.

Move left, Move right **Move left**  : Moves the chunk containing the currently selected box one tab-stop to the left.

Move right  : Moves the chunk containing the currently selected box one tab-stop to the right.

2.2.4 Menus

There are four menus, each explained on its own:

1. File menu
2. Edit menu
3. Tools menu
4. Help menu

File menu

Connect... The "Connect..." menu item connects to a database via a configuration file

equivalent ¡page_anchor id="1130"¿toolbar button:  .

Commit The "Commit" menu item saves the chunking-changes made in the program, by committing them to the database.

equivalent ¡page_anchor id="1130"¿toolbar button:  .

Exit The "Exit" menu item quits the program.
Equivalent toolbar button: None.

Edit menu

Split The "Split" menu item splits the current chunk right before the currently selected box.

Equivalent toolbar button:  .

Combine The "Combine" menu item combines the chunk containing the currently selected box with the previous chunk.

Equivalent toolbar button:  .

Move left The "Move left" menu item moves the chunk containing the currently selected box one tab-stop to the left.

Equivalent toolbar button:  .

Move right The "Move right" menu item moves the chunk containing the currently selected box one tab-stop to the right.

Equivalent toolbar button:  .

Tools menu

Configure... This menu-item has not been implemented yet. Sorry.

Help menu

Help Contents... This brings up this help document.

About Emdros Chunking Tool... This brings up the "About box". Press "OK" to dismiss it again.

Equivalent toolbar button: None.

2.2.5 Chunking Area

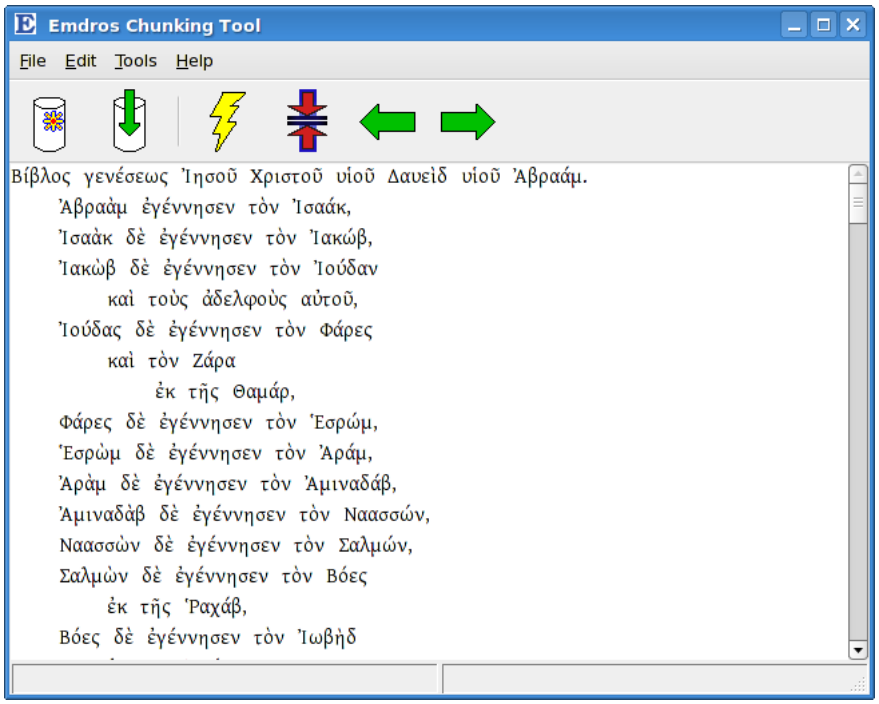
The chunking area is middle of the program window. It is this area that is the place for interacting with the program.

Once a database connection has been loaded, you will see some text in this window. The text is divided into clickable (and thus selectable) boxes.

You can use the "split"  and "combine"  buttons to split or combine the text into "chunks", each on its own line.

These chunks can then be indented with respect to each other with the "Move left"  and "Move right"  buttons.

Example In the image below, you can see an example database that has been chunked and indented.



3 Configuring the program

3.1

3.2 Format of the configuration file

The configuration file follows many other Unix and Windows configuration files in that:

- Comments are prefixed by #, and anything from the # to the end of the line is ignored.
- Blank lines are ignored.
- The rest is a number of "key = value" pairs.
- The keys are pre-defined (see below).
- The values are either "quote-enclosed strings" (e.g., "C:\Emdros\mymap.map") or consist of letters, numbers, underscores, and/or dots, optionally followed by a "quote-enclosed string" (e.g., 'word.surface', or 'word.surface."C:\Documents and Settings\Administrator\teckitmap.map"').

When a value has dots that are not enclosed in "quotes", then the strings on either side of the dots are interpreted as subkeys. For example, the value

"word.surface" represents the subkey "word" with the value "surface", and the value "word.surface."/home/myname/Blah.map" represents the subkey "word" with the subsubkey "surface", followed by the value "/home/myname/Blah.map".

Here is a sample configuration file, explained bit by bit:

3.2.1 Database selection

```
# database
database = mydb
```

You can specify a database that is always to be used with this configuration file.

If using SQLite, you may wish to specify a path. Do so in quotes:

```
# database
# Put the database name in quotes.
# For SQLite 2 and SQLite 3, you should probably give
# the full path to the file as well.

database = "C:\Program Files\Emdros\Emdros-1.2.0.pre228\db\mydb"
```

3.2.2 Data unit

```
# data unit
# There can only be one data unit
# but it can have as many data_features as you like.
# Each data_feature will go on its own interlinear line.
#

data_unit            = word
data_feature         = graphical_word
data_feature         = graphical_lexeme
```

The data unit is the basic unit that will result in one box in the chunking area. They can be any object type, and need not be words. However, probably you want them to be words or word-like objects. It depends on how large segments you want to be able to chunk at a time.

You must specify which feature(s) to display for the data unit.

There can only be one data unit.

3.2.3 TECkit mappings

```
# TECKit
#
```

```
# data_feature_teckit_mapping defines what TECkit map to use
# for a given data_feature.
#
# data_feature_teckit_in_encoding specifies the in_encoding ("bytes"
# or "unicode") for the given data_feature.
#
# data_feature_teckit_out_encoding specifies the out_encoding ("bytes"
# or "unicode") for the given data_feature.
#
data_feature_teckit_mapping      = graphical_word."Amsterdam.map"
data_feature_teckit_in_encoding  = graphical_word.bytes
data_feature_teckit_out_encoding = graphical_word.unicode
```

TECkit is a tool made by SIL International. It converts between encodings, in particular to and from Unicode. The Emdros Chunking Tool incorporates TECkit, and you can apply it to any textual feature of any object type.

TECkit works with a so-called "map file" – a text file which you or someone else writes. More information about writing TECkit mappings can be found on SIL's website:

http://scripts.sil.org/TECkit/

The Emdros Chunking Tool needs three pieces of information in order for TECkit to work on a particular feature:

1. The name of the file which holds the maping. This is given with the key "data_feature_teckit_mapping".
2. The input encoding (encoding of the feature-string): This is given with the key "data_feature_teckit_in_encoding". The value can be either "bytes" or "unicode" (without the quotes). "bytes" means that TECkit does not convert to UTF-8. "unicode" means it is converted to UTF-8 for display. You should use whatever is used in the map file for input encoding here.
3. The output encoding (encoding to transform into): This is given with the key "data_feature_teckit_out_encoding". The same meanings and restrictions apply as for the input encoding.

TECkit can not only convert between encodings, but also remove stuff from a string. This can come in handy when you have characters in your feature-strings which you do not wish to display. Again, see the TECkit site on SIL's website for information on how to write a TECkit mapping.

You should give first the object type, then a dot, then the feature-name, then a dot, then the full path to the map file. You probably need to enclose the path in "double quotes".

You can only have one TECkit per feature.

3.2.4 Options

```
# Options
```

```
#
# The only option available is 'right_to_left', which, if set,
# will cause the chunking area to run right to left rather than
# left to right.
option = right_to_left
```

3.2.5 Display options

```
# Fonts -- chunking area font names.
# If you give more than one chunking_area_font_name,
# they will be assigned to individual data_feature interlinear
# lines, in the same order as the data_feature keys appear.
#
# If you give less keys here than you have data_feature keys,
# then the last one will be used for the ones that aren't assigned
# an explicit value.
#
# If you give no values for this key, then some sensible default
# font will be used.
#
chunking_area_font_name = "Ezra SIL"
chunking_area_font_name = "Courier"
chunking_area_font_name = "Ezra SIL"

#
# The magnification (in percent) of the chunking area.
# 100 corresponds approximately to a font size of 12 points.
#
chunking_area_magnification  = 120
```