EDT Emulation User's Manual

Kevin Gallagher
kevin.gal@verizon.net

This file documents the EDT emulation package for Emacs.

Table of Contents

1 Overview of the EDT Package

This manual describes version 4.0 of the EDT Emulation for Emacs. It comes with special functions which replicate nearly all of EDT's keypad mode behavior. It sets up default keypad and function key bindings which closely match those found in EDT. Support is provided so that users may reconfigure most keypad and function key bindings to their own liking.

Version 4.0 contains several enhancements (see Section 1.2 [Changes], page 1).

1.1 How to Begin Using EDT

To start the EDT Emulation, first start Emacs and then enter *M-x* `edt-emulation-on` to begin the emulation. After initialization is complete, the following message will appear below the status line informing you that the emulation has been enabled: "Default EDT keymap active".

You can have the EDT Emulation start up automatically, each time you initiate an Emacs session, by adding the following line to your `.emacs` file:

```
(add-hook 'emacs-startup-hook 'edt-emulation-on)
```

Important: Be sure to read the rest of this manual. It contains very useful information on how the EDT Emulation behaves and how to customize it to your liking.

The EDT emulation consists of the following files:

- `edt.texi`—This manual.
- `edt-user.el`—An example customization file (located in the Emacs distribution etc directory).
- `edt.el`—EDT emulation functions and default configuration.
- `edt-lk201.el`—Built-in support for DEC LK-201 keyboards.
- `edt-vt100.el`—Built-in support for DEC VT-100 (and above) terminals.
- `edt-pc.el`—Built-in support for PC 101 Keyboards under MS-DOS.
- `edt-mapper.el`—Create an EDT LK-201 map file for keyboards without built-in support.

1.2 What's New in Version 4.0

Version 4.0 contains the following enhancements:

1. Scroll margins at the top and bottom of the window are now supported. (The design was copied from `tpu-extras.el`.) By default, this feature is enabled with the top margin set to 10% of the window and the bottom margin set to 15% of the window. To change these settings, you can invoke the function `edt-set-scroll-margins` in your `.emacs` file. For example, the following line

   ```
   (edt-set-scroll-margins "20%" "25%")
   ```

   sets the top margin to 20% of the window and the bottom margin to 25% of the window. To disable this feature, set each margin to 0%. You can also invoke `edt-set-scroll-margins` interactively while EDT Emulation is active to change the settings for that session.

Please note: Another way to set the scroll margins is to use the Emacs customization feature to set the following two variables directly: `edt-top-scroll-margin` and `edt-bottom-scroll-margin`.

Enter the Emacs `customize` command. First select the '`Editing`' group and then select the '`Emulations`' group. Finally, select the '`Edt`' group and follow the directions.

2. The '`SUBS`' command is now supported and bound to *`GOLD-Enter`* by default. (This design was copied from `tpu-edt.el`.) Note, in earlier versions of EDT Emulation, *`GOLD-Enter`* was assigned to the Emacs function `query-replace`. The binding of `query-replace` has been moved to *`GOLD-/`*. If you prefer to restore `query-replace` to *`GOLD-Enter`*, then use an EDT user customization file, `edt-user.el`, to do this (see Chapter 7 [Customizing], page 14).
3. If you access a workstation using an X Server, observe that the initialization file generated by `edt-mapper.el` will now contain the name of the X Server vendor. This is a convenience for those who have access to their Unix account from more than one type of X Server. Since different X Servers typically require different EDT emulation initialization files, `edt-mapper.el` will now generate these different initialization files and save them with different names. Then, the correct initialization file for the particular X server in use is loaded correctly automatically.
4. Also, `edt-mapper.el` is now capable of binding an ASCII key sequence, providing the ASCII key sequence prefix is already known by Emacs to be a prefix. As a result of providing this support, some terminal/keyboard/window system configurations, which don't have a complete set of sensible function key bindings built into Emacs in `input-decode-map`, can still be configured for use with EDT Emulation. (Note: In a few rare circumstances this does not work properly. In particular, it does not work if a subset of the leading ASCII characters in a key sequence are recognized by Emacs as having an existing binding. For example, if the keypad 7 (`KP7`) key generates the sequence '`ESCOw`' and '`ESCO`' is already bound to a function, pressing `KP7` when told to do so by `edt-mapper.el` will result in `edt-mapper.el` incorrectly mapping '`ESCO`' to `KP7` and '`w`' to `KP8`. If something like this happens to you, it is probably a bug in the support for your keyboard within Emacs **or** a bug in the Unix termcap/terminfo support for your terminal **or** a bug in the terminal emulation software you are using.)
5. The `edt-quit` function (bound to *`GOLD-q`* by default) has been modified to warn the user when file-related buffer modifications exist. It now cautions the user that those modifications will be lost if the user quits without saving those buffers.

1.3 The Aims of this Package

1. Emulate EDT Keypad Mode commands closely so that current EDT users will find that it easy and comfortable to use Emacs with a small learning curve.
2. Make it easy for a user to customize EDT emulation key bindings without knowing much about Emacs Lisp.
3. Make it easy to switch between the original EDT default bindings and the user's customized EDT bindings, without having to exit Emacs.
4. Provide support for some TPU/EVE functions not supported in EDT.
5. Provide an easy way to restore **all** original Emacs key bindings, just as they existed before the EDT emulation was first invoked.

6. Supports highlighting of marked text within the EDT emulation on all platforms on which Emacs supports highlighting of marked text.
7. Handle terminal configuration interactively for most terminal configurations, when the emulation is invoked for the first time.
8. Support a PC AT keyboard under MS-DOS.

2 Terminals/Keyboards that are Supported

Keyboards used under a Window System are supported via the `edt-mapper` function. The first time you invoke the emulation under a window system, the `edt-mapper` function is run automatically and the user is prompted to identify which keys the emulation is to use for the standard keypad and function keys EDT expects (e.g., `PF1`, `PF2`, `KP0`, `KP1`, `F1`, `F2`, etc.). This configuration is saved to disk read each time the emulation is invoked.

In character oriented connections not running a window manager, built-in support for the following terminals/keyboards is provided:

1. DEC VT-100 series and higher. This includes well behaved VT clones and emulators. If you are using a VT series terminal, be sure that the `TERM` environment variable is set properly before invoking emacs.
2. PC AT keyboard under MS-DOS.

Be sure to read Chapter 4 [Platform-specific notes], page 6, to see if those notes apply to you.

3 How to Get Started

Start up Emacs and enter *M-x edt-emulation-on* to begin the emulation. After initialization is complete, the following message will appear below the status line informing you that the emulation has been enabled: “Default EDT keymap active”.

You can have the EDT Emulation start up automatically, each time you initiate an Emacs session, by adding the following line to your `.emacs` file:

```
(add-hook 'emacs-startup-hook 'edt-emulation-on)
```

A reference sheet is included (later on) listing the default EDT Emulation key bindings. This sheet is also accessible on line from within Emacs by pressing `PF2`, *GOLD-H*, or ‘`HELP`’ (when in the EDT Default Mode).

It is easy to customize key bindings in the EDT Emulation (see Chapter 7 [Customizing], page 14). Customizations are placed in a file called `edt-user.el`. The Emacs `etc/` directory contains an example. If `edt-user.el` is found in your Emacs load path during EDT Emulation initialization, then the following message will appear below the status line indicating that the emulation has been enabled, enhanced by your own customizations: “User EDT custom keymap active”.

Once enabled, it is easy to switch back and forth between your customized EDT Emulation key bindings and the default EDT Emulation key bindings. (Look at the binding to *GOLD-Z* in the sample `edt-user.el` file.) It is also easy to turn off the emulation (via the command `edt-emulation-off`). Doing so completely restores the original key bindings in effect just prior to invoking the emulation.

Emacs binds keys to ASCII control characters and so does the real EDT. Where EDT key bindings and Emacs key bindings conflict, the default Emacs key bindings are retained by the EDT emulation by default. If you are a diehard EDT user you may not like this. The Section 7.3 [Control keys], page 15, section explains how to change this so that the EDT bindings to ASCII control characters override the default Emacs bindings.

4 Notes Specific to Certain Platforms

4.1 Sun Workstations Running X

Some earlier Sun keyboards do not have arrow keys separate from the keypad keys. It is difficult to emulate the full EDT keypad and still retain use of the arrow keys on such keyboards.

The Sun Type 5 and other more recent Sun keyboards, however, do have separate arrow keys. This makes them candidates for setting up a reasonable EDT keypad emulation.

Depending upon the configuration of the version of X installed on your system, you may find the default X keynames for the keypad keys don't permit Emacs to interpret some or all the keypad keys as something other than arrow keys, numeric keys, `Home`, `PageUp`, etc. Both Sun and HP have been particularly guilty of making bizarre keysym assignments to the keypad keys.

In most cases, the X Windows command, `xmodmap`, can be used to correct the problem. Here's a sample `.xmodmaprc` file which corrects this problem on one Sun workstation configuration using an older SunOS release configured with a Sun Type 5 keyboard:

```
! File:  .xmodmaprc
!
! Set up Sun Type 5 keypad for use with the Emacs EDT Emulation
!
keycode  53 = KP_Divide
keycode  54 = KP_Multiply
keycode  57 = KP_Decimal
keycode  75 = KP_7
keycode  76 = KP_8
keycode  77 = KP_9
keycode  78 = KP_Subtract
keycode  97 = KP_Enter
keycode  98 = KP_4
keycode  99 = KP_5
keycode 100 = KP_6
keycode 101 = KP_0
keycode 105 = F24
keycode 119 = KP_1
keycode 120 = KP_2
keycode 121 = KP_3
keycode 132 = KP_Add
```

If `edt-mapper.el` does not recognize your keypad keys as unique keys, use the command '`xmodmap -pke`' to get a listing of the actual key codes and the keysyms mapped to them and then generate you own custom `.xmodmaprc` similar to the one above.

Next, feed `.xmodmaprc` to the `xmodmap` command and all the Sun Type 5 keypad keys will now be configurable for the emulation of an LK-201 keypad (less the `,` key). In this example, the line

```
keycode 105 = F24
```

changes the X Windows name of the keypad `NumLock` key to be known internally as the `F24` key. Doing so permits it to be configured to behave as the `PF1` (`GOLD`) key.

The side effect of this change is that you will no longer have a `NumLock` key. If you are using other software under X which requires a `NumLock` key, then examine your keyboard and look for one you don't use and redefine it to be the `NumLock` key. Basically, you need to clear the `NumLock` key from being assigned as a modifier, assign it to the key of your choice, and then add it back as a modifier. (Section 4.4 [Unix], page 7, for further help on how to do this.)

4.2 PC Users Running MS-DOS

By default, F1 is configured to emulate the `PF1` (`GOLD`) key. But `NumLock` can be used instead if you load a freeware TSR distributed with MS-Kermit, call '`gold.com`'. This was once distributed in a file called `gold22.zip` and came with the source code as well as a loadable binary image. (See `edt-pc.el` in the Emacs `lisp/emulation` directory for more information.)

4.3 PC Users Running GNU/Linux

The default X server configuration varies from distribution to distribution and release to release of GNU/Linux. If your system fails to recognize the keypad keys as distinct keys, change the NumLock state, turning it on or off, as the case may be, then try again. If this doesn't solve your problem, you may have to modify the X keysym mappings with `xmodmap`.

On one distribution on an Intel PC, the following `.xmodmaprc` set things up nicely.

```
! File:  .xmodmaprc
!
! Set up PC keypad under GNU/Linux for the Emacs EDT Emulation
!
clear  mod2
keycode  77 = F12
keycode  96 = Num_Lock Pointer_EnableKeys
add mod2 = Num_Lock
```

In this example, after feeding the file to the `xmodmap` command, the PC `NumLock` keypad key will be configurable for the emulation of the `PF1` key. The PC keypad can now emulate an LK-201 keypad (less the comma key), the standard keyboard supplied with DEC terminals VT-200 and above. This `.xmodmaprc` file switches the role of the `F12` and `NumLock` keys. It has been tested on RedHat GNU/Linux 5.2. Other versions of GNU/Linux may require different keycodes. (Section 4.4 [Unix], page 7, for further help on how to do this.)

Please note: Remember, it may be necessary to have `NumLock` in one position (ON) or the other (OFF) for the PC keypad to emulate the LK-201 keypad properly.

4.4 General Notes on Using `NumLock` for the `PF1` Key on Unix Systems

Making the physical `NumLock` key available for use in the EDT Emulation requires some modification to the default X Window settings. Since the keycode assignments vary from system to system, some investigation is needed to see how to do this on a particular system.

You will need to look at the output generated by xmodmap invoked with the "-pm" switch. For example, on RedHat GNU/Linux 5.2 on a PC, we get the following output when running ‘xmodmap -pm’:

```
xmodmap:  up to 2 keys per modifier, (keycodes in parentheses):

shift       Shift_L (0x32),  Shift_R (0x3e)
lock        Caps_Lock (0x42)
control     Control_L (0x25),  Control_R (0x6d)
mod1        Alt_L (0x40),  Alt_R (0x71)
mod2        Num_Lock (0x4d)
mod3
mod4
mod5        Scroll_Lock (0x4e)
```

Note that Num_Lock is assigned to the modifier ‘mod2’. This is what hides Num_Lock from being seen by Emacs.

Now, ‘xmodmap -pke’ yields:

```
    .
    .
    .
keycode  77 = Num_Lock Pointer_EnableKeys
    .
    .
    .
keycode  96 = F12
    .
    .
    .
```

So, in RedHat GNU/Linux 5.2 on a PC, Num_Lock generates keycode 77. The following steps are taken:

1. clear the assignment of Num_Lock to mod2;
2. swap the keycodes assigned to F12 and Num_Lock;
3. assign Num_Lock back to mod2.

The .xmodmaprc file looks like this:

```
! File:  .xmodmaprc
!
! Set up PC keypad under GNU/Linux for the Emacs EDT Emulation
!
clear  mod2
keycode  77 = F12
keycode  96 = Num_Lock Pointer_EnableKeys
add mod2 = Num_Lock
```

So, after executing ‘xmodmap .xmodmaprc’, a press of the physical F12 key looks like a Num_Lock keypress to X. Also, a press of the physical NumLock key looks like a press of the F12 key to X.

Now, `edt-mapper.el` will see '`f12`' when the physical `NumLock` key is pressed, allowing the `NumLock` key to be used as the EDT `PF1` (`GOLD`) key.

5 How Does this EDT Emulation Differ from Real EDT?

In general, you will find that this emulation of EDT replicates most, but not all, of EDT's most used Keypad Mode editing functions and behavior. It is not perfect, but most EDT users who have tried the emulation agree that it is quite good enough to make it easy for die-hard EDT users to move over to using Emacs.

Here's a list of the most important differences between EDT and this GNU Emacs EDT Emulation. The list is short but you must be aware of these differences if you are to use the EDT Emulation effectively.

1. Entering repeat counts works a little differently than in EDT.

   EDT allows users to enter a repeat count before entering a command that accepts repeat counts. For example, when using the real EDT, pressing these three keys in sequence, *GOLD 5 KP1*, will move the cursor in the current direction 5 words. This does **not** work in Emacs!

   Emacs provides two ways to enter repeat counts and neither involves using the GOLD key. First, repeat counts can be entered in Emacs by using the ESC key. For example, pressing these keys in sequence, *ESC 1 0 KP1*, will move the cursor in the current direction 10 words. Second, Emacs provides another command called universal-argument that can be used to do the same thing. Normally, in Emacs has this bound to *C-u*.

2. EDT's line mode commands and nokeypad mode commands are **not** supported (with one important exception; see item 8 in Chapter 6 [Highlights], page 11). Although, at first, this may seem like a big omission, the set of built-in Emacs commands provides a much richer set of capabilities which more than make up for this omission.

   To enter Emacs commands not bound to keys, you can press *GOLD KP7* or the DO key. Emacs will display its own command prompt "M-x". This stands for the keypress *Meta-x*, where Meta is a special shift key. The Alt key is often mapped to behave as a Meta key. So, you can also invoke this prompt by pressing *Meta-x*. Typing the sequence *ESC x* will also invoke the prompt.

3. Selected text is highlighted **only** on systems where Emacs supports the highlighting of text.

4. Just like in TPU/EVE, the ENTER key is **not** used to terminate input when the editor prompts you for input. The RETURN key is used, instead. (KP4 and KP5 (the direction keys) do terminate input for the 'FIND' command, just like in EDT, however.)

6 Some Highlights, and Comparisons to the Original Emacs EDT Emulation

1. The EDT define key command is supported (`edt-define-key`) and is bound to *`C-k`* in the default EDT mode when EDT control sequence bindings are enabled, or when the sample `edt-user.el` customization file is used. The TPU/EVE learn command is supported but not bound to a key in the default EDT mode but is bound in the sample `edt-user.el` file.

   Unlike the TPU/EVE learn command, which uses one key to begin the learn sequence, *`C-l`*, and another command to remember the sequence, *`C-r`*, this version of the learn command (`edt-learn`) serves as a toggle to both begin and to remember the learn sequence.

   Many users who change the meaning of a key with the define key and the learn commands, would like to be able to restore the original key binding without having to quit and restart emacs. So a restore key command is provided to do just that. When invoked, it prompts you to press the key to which you wish the last replaced key definition restored. It is bound to *`GOLD C-k`* in the default EDT mode when EDT control sequence bindings are enabled or the sample `edt-user.el` customization file is used.
2. Direction support is fully supported.
3. All original Emacs bindings are fully restored when EDT emulation is turned off. So, if a fellow worker comes over to your terminal to help you with a software problem, for example, and is completely confused by your EDT emulation bindings, just enter the command, `edt-emulation-off`, at the '`M-x`' prompt and the original Emacs bindings will be restored. To resume the EDT emulation, just enter `edt-emulation-on`.
4. User custom EDT bindings are kept separate from the default EDT bindings. One can toggle back and forth between the custom EDT bindings and default EDT bindings.
5. The Emacs functions in `edt.el` attempt to emulate, where practical, the exact behavior of the corresponding EDT keypad mode commands. In a few cases, the emulation is not exact, but we hope you will agree it is close enough. In a very few cases, we chose to use the Emacs way of handling things. As mentioned earlier, we do not emulate the EDT '`SUBS`' command. Instead, we chose to use the Emacs `query-replace` function, which we find to be easier to use.
6. Emacs uses the regexp assigned to `page-delimiter` to determine what marks a page break. This is normally '`^\f`', which causes the `edt-page` command to ignore form feeds not located at the beginning of a line. To emulate the EDT '`PAGE`' command exactly, page-delimiter is set to '`\f`' when EDT emulation is turned on, and restored to '`^\f`' when EDT emulation is turned off. But, since some users prefer the Emacs definition of a page break, or may wish to preserve a customized definition of page break, one can override the EDT definition by placing

   ```
   (setq edt-keep-current-page-delimiter t)
   ```

   in your `.emacs` file. Or, you can used the Emacs customize command to change its setting.
7. The EDT definition of a section of a terminal window is hardwired to be 16 lines of its one-and-only 24-line window (the EDT '`SECT`' command bound to `KP8`). That's two-thirds of the window at a time. Since Emacs, like TPU/EVE, can handle multiple windows of sizes of other than 24 lines, the definition of section used here has been

modified to two-thirds of the current window. (There is also an `edt-scroll-window` function which you may prefer over the '`SECT`' emulation.)

8. Cursor movement and deletion involving word entities is identical to EDT. This, above all else, gives the die-hard EDT user a sense of being at home. Also, an emulation of EDT's '`SET ENTITY WORD`' command is provided, for those users who like to customize movement by a word at a time to their own liking.
9. EDT's '`FIND`' and '`FNDNXT`' are supported.
10. EDT's '`APPEND`', '`REPLACE`', and '`SUBS`' commands are supported.
11. '`CHNGCASE`' is supported. It works on individual characters or selected text, if '`SELECT`' is active. In addition, two new commands are provided: `edt-lowercase` and `edt-uppercase`. They work on individual **words** or selected text, if '`SELECT`' is active.
12. Form feed and tab insert commands are supported.
13. A new command, `edt-duplicate-word`, is provided. If you experiment with it, you might find it to be surprisingly useful and may wonder how you ever got along without it! It is assigned to *`C-j`* in the sample `edt-user.el` customization file.
14. TPU/EVE's Rectangular Cut and Paste functions (originally from the EVE-Plus package) are supported. But unlike the TPU/EVE versions, these here support both insert and overwrite modes. The seven rectangular functions are bound to `F7`, `F8`, *`GOLD-F8`*, `F9`, *`GOLD-F9`*, `F10`, and *`GOLD-F10`* in the default EDT mode.
15. The original EDT emulation package set up many default regular and GOLD bindings. We tried to preserve most (but not all!) of these, so users of the original emulation package will feel more at home.

    Nevertheless, there are still many GOLD key sequences which are not bound to any functions. These are prime candidates to use for your own customizations.

    Also, there are several commands in `edt.el` not bound to any key. So, you will find it worthwhile to look through `edt.el` for functions you may wish to add to your personal customized bindings.
16. The VT200/VT300 series terminals steal the function keys `F1` to `F5` for their own use. These do not generate signals which are sent to the host. So, `edt.el` does not assign any default bindings to `F1` through `F5`.

    In addition, our VT220 terminals generate an interrupt when the `F6` key is pressed ('`^C`' or '`^Y`', can't remember which) and not the character sequence documented in the manual. So, binding Emacs commands to `F6` will not work if your terminal behaves the same way.
17. The VT220 terminal has no `ESC`, `BS`, nor `LF` keys, as does a VT100. So the default EDT bindings adopt the standard DEC convention of having the `F11`, `F12`, and `F13` keys, on a VT200 series (and above) terminal, assigned to the same EDT functions that are bound to `ESC`, `BS`, and `LF` on a VT100 terminal.
18. Each user, through the use of a private `edt-user.el` file, can customize, very easily, personal EDT emulation bindings.
19. The EDT '`SELECT`' and '`RESET`' functions are supported. However, unlike EDT, pressing '`RESET`' to cancel text selection does **not** reset the existing setting of the current direction.

We also provide a TPU/EVE like version of the single '`SELECT/RESET`' function, called `edt-toggle-select`, which makes the EDT '`SELECT`' function into a toggle on/off switch. That is, if selection is on, pressing '`SELECT`' again turns selection off (cancels selection). This function is used in the sample `edt-user.el` customization file.

20. EDT scroll margins are supported, but are disabled by default. (Section 7.4 [Scroll margins], page 15, for instructions on how to enable them.)

7 Customizing Emulation

Most EDT users, at one time or another, make some custom key bindings, or use someone else's custom key bindings, which they come to depend upon just as if they were built-in bindings. This EDT Emulation for GNU Emacs is designed to make it easy to customize bindings.

If you wish to customize the EDT Emulation to use some of your own key bindings, you need to make a private version of `edt-user.el` in your own private lisp directory. The Emacs `etc/` directory contains an example for you to use as a template and for ideas.

First, you need to have your own private lisp directory, say `~/lisp`, and you should add it to the Emacs load path.

Please note: A few sites have different load-path requirements, so the above directions may need some modification if your site has such special needs.

7.1 Creating your own `edt-user.el` File

A sample `edt-user.el` file is provided in the Emacs `etc/` directory. You should use it as a guide to learn how you can customize EDT emulation bindings to your own liking. Names used to identify the set of LK-201 keypad and function keys are:

```
Keypad Keys:
                PF1 PF2 PF3 PF4
                KP7 KP8 KP9 KP-
                KP4 KP5 KP6 KP,
                KP1 KP2 KP3
                KP0     KPP KPE
Arrow Keys:
                LEFT RIGHT DOWN UP
Function Keys:
                F1 F2 F3 F4 F5  F6 F7 F8 F9 F10  F11 F12 F13 F14
                HELP DO  F17 F18 F19 F20

                FIND   INSERT   REMOVE
                SELECT PREVIOUS NEXT
```

Note: Many VT-200 terminals, and above, steal function keys `F1` through `F5` for terminal setup control and don't send anything to the host if pressed. So customizing bindings to these keys may not work for you.

There are three basic functions that do the EDT emulation custom bindings: `edt-bind-key`, `edt-bind-gold-key`, and `edt-bind-function-key`.

The first two are for binding functions to keys which are standard across most keyboards. This makes them keyboard independent, making it possible to define these key bindings for all terminals in the file `edt.el`.

The first, `edt-bind-key`, is used typically to bind emacs commands to control keys, although some people use it to bind commands to other keys, as well. (For example, some people use it to bind the VT200 seldom used back-tick key ('`'') to the function '`ESC-prefix`' so it will behave like an `ESC` key.) The second function, `edt-bind-gold-key`, is used to bind emacs commands to gold key sequences involving alpha-numeric keys, special character keys, and control keys.

The third function, `edt-bind-function-key`, is terminal dependent and is defined in a terminal specific file (see `edt-vt100.el` for example). It is used to bind emacs commands to LK-201 function keys, to keypad keys, and to gold sequences of those keys.

7.2 Specifying Word Entities

The variable `edt-word-entities` is used to emulate EDT's '`SET ENTITY WORD`' command. It contains a list of characters to be treated as words in themselves. If the user does not define `edt-word-entities` in his/her `.emacs` file, then it is set up with the EDT default containing only `TAB`.

The characters are stored in the list by their numerical values, not as strings. Emacs supports several ways to specify the numerical value of a character. One method is to use the question mark: '`?A`' means the numerical value for '`A`', '`?/`' means the numerical value for '`/`', and so on. Several unprintable characters have special representations:

```
?\b  specifies  BS, C-h
?\t  specifies  TAB, C-i
?\n  specifies  LFD, C-j
?\v  specifies  VTAB, C-k
?\f  specifies  FF, C-l
?\r  specifies  CR, C-m
?\e  specifies  ESC, C-[
?\\  specifies  \
```

Here are some examples:

```
(setq edt-word-entities '(?\t ?- ?/)) ; specifies TAB, - , and /
(setq edt-word-entities '(?\t)        ; specifies TAB, the default
```

You can also specify characters by their decimal ascii values:

```
(setq edt-word-entities '(9 45 47))   ; specifies TAB, - , and /
```

7.3 Enabling EDT Control Key Sequence Bindings

Where EDT key bindings and Emacs key bindings conflict, the default Emacs key bindings are retained by default. Some diehard EDT users may not like this. So, if the variable `edt-use-EDT-control-key-bindings` is set to true in a user's `.emacs` file, then the default EDT Emulation mode will enable most of the original EDT control key sequence bindings. If you wish to do this, add the following line to your `.emacs` file:

```
(setq edt-use-EDT-control-key-bindings t)
```

7.4 Setting Scroll Margins

Scroll margins at the top and bottom of the window are now supported. (The design was copied from `tpu-extras.el`.) By default, this feature is enabled with the top margin set to 10% of the window and the bottom margin set to 15% of the window. To change these settings, you can invoke the function `edt-set-scroll-margins` in your `.emacs` file. For example, the following line

```
(edt-set-scroll-margins "20%" "25%")
```

sets the top margin to 20% of the window and the bottom margin to 25% of the window. To disable this feature, set each margin to 0%. You can also invoke `edt-set-scroll-margins` interactively while EDT Emulation is active to change the settings for that session.

Please note: Another way to set the scroll margins is to use the Emacs customization feature to set the following two variables directly: `edt-top-scroll-margin` and `edt-bottom-scroll-margin`.

Enter the Emacs `customize` command. First select the '`Editing`' group and then select the '`Emulations`' group. Finally, select the '`Edt`' group and follow the directions.

Appendix A GNU Free Documentation License

Version 1.3, 3 November 2008

0. PREAMBLE

   The purpose of this License is to make a manual, textbook, or other functional and useful document *free* in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

   This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

   We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

   This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

   A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

   A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

   The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

The "publisher" means any person or entity that distributes copies of the Document to the public.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

   If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

   If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

   If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

   It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

   You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

   A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

C. State on the Title page the name of the publisher of the Modified Version, as the publisher.

D. Preserve all the copyright notices of the Document.

E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this License.

I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.

N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.

O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

   You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

   The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

   In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements."

6. COLLECTIONS OF DOCUMENTS

   You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

   You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

   A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

   If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

   Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

   If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

   You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

   However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

   Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

   Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

10. FUTURE REVISIONS OF THIS LICENSE

    The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See `https://www.gnu.org/licenses/`.

    Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

11. RELICENSING

    "Massive Multiauthor Collaboration Site" (or "MMC Site") means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A "Massive Multiauthor Collaboration" (or "MMC") contained in the site means any set of copyrightable works thus published on the MMC site.

    "CC-BY-SA" means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

    "Incorporate" means to publish or republish a Document, in whole or in part, as part of another Document.

    An MMC is "eligible for relicensing" if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

    The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

```
Copyright (C)  year  your name.
Permission is granted to copy, distribute and/or modify this document
under the terms of the GNU Free Documentation License, Version 1.3
or any later version published by the Free Software Foundation;
with no Invariant Sections, no Front-Cover Texts, and no Back-Cover
Texts.  A copy of the license is included in the section entitled ``GNU
Free Documentation License''.
```

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with. . . Texts." line with this:

```
with the Invariant Sections being list their titles, with
the Front-Cover Texts being list, and with the Back-Cover Texts
being list.
```

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.