

# EOL Changer

## Introduction

In this era of incompatible operating systems we often receive a text file, which could be a computer program, with different line-ending marks . These line-ending marks are sometimes called EOLs (End Of Lines.)

To be precise, DOS and Windows computers use an EOL consisting of a **CR** followed by an **LF**. Macintosh computers use a **CR** alone. Unix and Linux systems use an **LF** alone. (In the ASCII scheme of things, a **CR** is character 13 and an **LF** is character 10. True BASIC can generate these with a `chr$(13)` and a `chr$(10)`.)

The rest of this note is a computer program that changes the EOLs for a text file. It should be easy to use. To use it, first “copy” from this **pdf** document, then “paste” into a True BASIC editing window. The program includes a portion of the True Dials library that comes with the True BASIC Silver and Gold Editions. (You will have to edit a bit as some of the True BASIC code lines are too long to fit on this page without “folding.”) Since it is completely self-contained, you can examine it to see how it works, and make changes as you wish.

## The Program

```
! EOLChanger.tru

OPTION TYPO

LOCAL FromType, ToType, wineol$, maceol$, unixeol$
LOCAL fromeol$, lenfromeol, toeol$, lentoeol, r
LOCAL fromfilename$, tofilename$, n, s$, p, lenfromeolml, filetype

DO
    CALL TD_Warn ("From type EOL:", "DOS-Windows|Mac|Unix-Linux", 0, FromType)
    CALL TD_Warn ("To type EOL:", "DOS-Windows|Mac|Unix-Linux", 0, ToType)

    IF FromType = ToType then
        CALL TD_Warn ("Types cannot be the same; try again", "Continue", 1,
result)
    ELSE
        EXIT DO
    END IF

LOOP

LET wineol$ = chr$(13) & chr$(10)
LET maceol$ = chr$(13)
LET unixeol$ = chr$(10)

SELECT CASE FromType
CASE 1
    LET fromeol$ = wineol$
    LET lenfromeol = 2
CASE 2
```

```

        LET fromeol$ = maceol$
        LET lenfromeol = 1
CASE 3
        LET fromeol$ = unixeol$
        LET lenfromeol = 1
END SELECT

SELECT CASE ToType
CASE 1
        LET toeol$ = chr$(13) & chr$(10)
        LET lentoeol = 2
CASE 2
        LET toeol$ = chr$(13)
        LET lentoeol = 1
CASE 3
        LET toeol$ = chr$(10)
        LET lentoeol = 1
END SELECT

LET lenfromeolm1 = lenfromeol - 1

DO
    CALL TD_GetFile ("", fromfilename$, 1)
    IF fromfilename$ = "" then STOP

    CALL TD_SaveFile ("", tofilename$)
    IF tofilename$ = "" then STOP

    OPEN #1: name fromfilename$, org byte
    ASK #1: FILESIZE n
    READ #1, bytes n: s$

    OPEN #2: name tofilename$, org byte, create newold
    ERASE #2

    IF pos(s$, wineol$) > 0 then
        LET filetype = 1
    ELSE IF pos(s$, maceol$) > 0 then
        LET filetype = 2
    ELSE IF pos(s$, unixeol$) > 0 then
        LET filetype = 3
    END IF

    IF FromType <> filetype then
        CALL TD_Warn ("From file not the type you think it is.", "Stop", 1, r)
        STOP
    END IF

    LET p = 0
    DO
        LET p = pos(s$, fromeol$, p)
        IF p = 0 then EXIT DO
        LET s$[p:p+lenfromeolm1] = toeol$

```

```

        LET p = p + lentoeol
    LOOP
    WRITE #2: s$

    CALL TD_Warn ("Another file?", "Yes|No", 2, r)
    IF r = 2 then STOP

LOOP

END

! MODULE TrueDials
! Version 5.0
! Copyright (c) 1994, 1995, and 1998 by True BASIC, Inc.
! All rights reserved

MODULE TrueDials

    OPTION TYPO

    ! SUB TD_Warn (message$, button$, default, result)
    ! SUB TD_Message (title$, message$, button$, default, result)
    ! SUB TD_GetFile (extension$, filename$, changedir)
    ! SUB TD_SaveFile (extension$, filename$)

PRIVATE Breakup, Concat

SHARE dbox_timeout, xloc, yloc
LET dbox_timeout = 0          ! 0 means no timeout
LET xloc, yloc = -1

! For all the routines, up to three buttons can be displayed.
! The text for these is provided in the string button$, separated
! by "|". Default gives the number of the outlined button.
! The number of the button selected is returned in result.

! TD_SetLocation
!
! Allows setting the upper-left corner to override the default
!
SUB TD_SetLocation (x, y)

    LET xloc = x
    LET yloc = y

END SUB

! TD_Warn
!
! Displays warning message on the screen at a central position.
!
SUB TD_Warn (message$, button$, default, result)

```

