

## Dialog box: Configuration - Change button bar

**Hint:** By dragging a file on the button bar **with pressed SHIFT key** you can easily add buttons to the button bar. By pressing the **right** mouse button on the button bar, you can open a local menu which allows you to change or delete the button under the mouse cursor. The **Change button** dialog box has exactly the same structure as the lower part of the **Change button bar** dialog box.

**New in Total Commander 11:** When you change a button bar, all other currently running instances of Total Commander will also be informed to update their bar.

Field	Description
<b><u>B</u>utton bar:</b>	Behind this identifier the file name of the actual button bar is shown. Behind the file name, you can choose another button bar using the >> button. You cannot change the bar name if you are in a <b>subbar</b> . You can only change the name of the default bar (loaded at startup). Below this identifier you can see the actual button bar in a raw form (only the icons). <b>Hint:</b> You can use <b>SHIFT+arrow keys</b> to move buttons or drag them around with your mouse!
<b><u>S</u>ize:</b>	Here you can choose a size for the buttons in the button bar. A size of 32 corresponds to the normal icon size. New with TC 9.20, the size will be stored separately for each screen scale factor (dpi).
<b><u>X</u>P-Style</b>	Uses the current theme to draw the buttons (Windows XP or newer)
<b><u>F</u>lat icons</b>	If checked the icons are shown without 3d effect. Supports themed toolbar buttons.
<b><u>S</u>mall icons</b>	If checked, the icons are resized to the given size (default: 16x16), otherwise they are resized to the button size.
<b><u>A</u>dd...</b>	Adds a new entry in the button bar directly BEHIND the current position.
<b><u>D</u>elete</b>	Deletes the currently selected icon.
<hr/>	
<b><u>C</u>ommand:</b>	<p>There are many possible commands which can be placed in this field:</p> <ol style="list-style-type: none"> <li>You may specify the program to be run, including its search path. You can also specify fixed parameters here. <b>Important:</b> You must give the extensions <b>.com</b> and <b>.bat</b>, because by default Windows only finds <b>.exe</b>! With <b>"Change &gt;&gt;"</b> and <b>"Insert file name"</b> you can search for the desired program. Put an asterisk '*' in front of the name (full path required) to start it with administrative rights, or two '*' to start it as a different user (Windows 7 or newer). Supports <b>environment variables</b>. <b>Warning:</b> If Total Commander is not run as an administrator, be aware of the following issues when running programs as an administrator: <ul style="list-style-type: none"> <li>* Environment variables specific to Total Commander (like %COMMANDER_PATH%) will not be inherited by the programs.</li> <li>* In most Windows versions, the programs will not run in the folder indicated by the "Start path" field.</li> <li>* If the program is a batch file (.bat or .cmd file), you may need to precede it with cmd.exe /c.</li> </ul> </li> <li>You can also place one or more of Total Commander's internal menu commands here comma-separated. Just click on the button with the magnifying glass and choose a command from the dialog box <b>"Choose command"</b>. Can be combined with cd command (change directory) with the path in double quotes. Example for multiple commands: cd "c:\windows",cm_SelectAll,cm_Copy cm_wait x adds a delay of x milliseconds, e.g. cm_SelectAll,cm_wait 1000,cm_Copy You can open a virtual folder with one of the following commands: cm_OpenDesktop, cm_OpenDrives, cm_OpenControls, cm_OpenFonts, cm_OpenNetwork, cm_OpenPrinters, cm_OpenRecycled Instead of the command name, you can also use the command number as listed in totalcmd.inc, e.g. 300 instead of cm_SrcComments. Additionally, you can define your own commands with prefix em_ via <b>"Choose command"</b> and then use them here.</li> </ol>

- Many internal commands support a single parameter, see "[Choose command](#)" for a list.
3. You can make a button to change directories, by specifying **cd *directory name*** as the command. You can even choose which files to show, e.g. using the command **cd *directory\\*.txt***, but not when combining the cd command with other commands.  
With parameter **"/user:"** (all lowercase!) you can specify the user to use for the connection to the remote share. If no user name is given, you will be asked to enter a user name and password.  
Also supports virtual folders, e.g. **cd shell:Fonts** or **cd ::{GUID of folder}**  
**Example:** **cd \\server\share /user:username**
  4. You can also open an ftp connection, using either **cd ftp://ftp.servername.com**  
or: **FTPOPEN *session\_name***, where *session\_name* is the name configured in the [FTP connect dialog box](#).  
or: **FTPOPEN *session\_name*/subdir**, connects to session (in preferred file panel) if not yet connected, and goes to specified subdirectory  
or **FTPOPEN *session\_name*\*/subdir**, same but always in active panel, e.g. for Ctrl+D.
  5. With the command **ZIPFROMLIST**, you can create a zip file from a list of files. The command field must contain the zip file name, and the parameters field the name of the list file.  
Alternatively, the zip file name may be put at the start of the parameters field, surrounded by double quotes "" if it contains a space. The list file must contain a list of files to be packed, one file per line (ANSI, UTF-16 or UTF-8, with byte order marker BOM). A parameter like %F may be used for the selected files in the active panel.  
Supports the following parameters, put in front of the zip file name (only in the field 'Command'):  
-m move  
-o overwrite without asking  
-os Skip all  
-on Overwrite all older  
-0 .. -9 compression rate  
-c0 encrypt (zip20), -c1 AES128, -c2 AES192, -c3 AES256  
-c0pass .. -c3pass encrypt with given password
  6. As a further possibility, you can insert a button to **switch to another button bar**. To do this, click on the button **Add subbar >>**. In the appearing dialog box you can choose an existing bar as a subbar or enter a name for a new bar. After this, another bar appears, which allows you to change the subbar. You can make circular references (bar 1 calls bar 2 calls bar 3), but avoid to edit a bar for which you have already a window opened!  
**Note:** If you want to have another bar at this position, you must delete the file name of the old bar before pressing the **Add subbar >>** button.
  7. **APPENDTABS *tabfile.tab*** appends the [folder tabs](#) stored in the file *tabfile.tab* to the current tab header. Uses "file open" dialog box when no parameter is given.  
To append tabs on the left/right side, append the character **L** or **R** to the command, respectively.
  8. **OPENTABS *tabfile.tab*** same, but replaces the current tabs with these new tabs.  
To open tabs on the left/right side, append the character **L** or **R** to the command, respectively.
  9. **SYNCOPE saved\_sync\_name** opens synchronize dirs with saved options. If the parameter begins with an equal sign "=", the comparison will start immediately. If the equal sign is the only parameter, the comparison will start with the current directories and last used options.
  10. **LOADSEARCH saved\_search\_name** opens search function with saved search. An equal sign "=" in front of the name will start the search immediately, two equal signs will show the results directly in Total Commander.
  11. **OPENBAR saved\_bar** loads an existing button bar, for example default.bar.
  12. **MULTIRENAME saved\_rename\_settings** opens the multi-rename tool with the specified settings. An equal sign "=" in front of the name will start the renaming immediately if the settings can be loaded, two equal signs will close the multi-rename tool if there were no errors.
  13. **OPENCUSTOMVIEW custom\_columns\_view\_title** switches to the first custom columns view with the specified name. The name is case-sensitive. To set the view on a specific side, append one of the suffixes **L**, **R**, **S**, **T** or **B** to the command (left/right/source/target/both), e.g. **OPENCUSTOMVIEWL my\_view** to switch the view in the left panel.
  14. **SELECTFILES file types** marks files of the given type, e.g. **SELECTFILES \*.txt \*.doc**  
Supported modifiers: **D**=only folders, **B**=both, **U**=unselect, **L**=left side, **R**=right side, **T**=target panel, **S**=When Shift is pressed during execution, marks also folders. Add directly to the end

of the command name, like **selectfilesD** or **selectfilesDUT** or any meaningful combination.

**15.** A value of -2 and no icon causes a manual line break in the button bar

**16. SAVETABS, SAVETABSL, SAVETABSR, SAVETABS2, SAVETABS2L, SAVETABS2R:** save tabs to a specified file. The first 3 save just one side (active, left, right), the second 3 save both sides (active first, left first, right first).

**17. SAVESELECTION, SAVESELECTIONA, SAVESELECTIONW:** save names of selected files to the specified file. With character A at the end, the list is saved as ANSI, with W as Unicode UTF-16. Without suffix, it depends on the operating system (NT-based systems use Unicode).

**18. SAVEDETAILS, SAVEDETAILSA, SAVEDETAILSW:** save names and details of selected files to the specified file. With character A at the end, the list is saved as ANSI, with W as Unicode UTF-16. Without suffix, it depends on the operating system (NT-based systems use Unicode). Append H after DETAILS to save with headers.

**19. LOADSELECTION:** load selection of files from a specified file. Supports ANSI and Unicode UTF-16.

**20. OPENLANGUAGEFILE:** load specified language file, e.g. wcmd\_deu.lng, from the language subfolder. No parameter = internal English.

**21. LOADLIST:** load list of files/folders from the specified text file. There must be one name per line. If no path is provided, the current directory is assumed. File details are retrieved automatically. Wildcards can also be used, but regular expressions are not supported.

**22. LOADLIST0:** Like LOADLIST, but do not show any error messages if files cannot be found.

**23. OPENATTRIBUTES:** Open attribute options with **specific parameters**

**24. OPENBARMENU saved\_bar** loads an existing button bar as a dropdown menu. To open a .bar file from the current directory, the name must be provided in the following way:  
.\filename.bar

#### Start path:

Determines the path to be set before the program is started. If no path is given, then the path of the source window is set.

**Important:** If the command (given in the command box) is **cd drive:\directory**, then the path given in this edit box is set in the **other window!** This allows you to set both paths (source and target) at the same time.

#### Parameters:

Here you can specify command line parameters. Fixed parameters **must** be specified directly after the file name, because otherwise, in a Drag&Drop, only the file name would be given to the program.

**Note:** All parameters now support substring fields in the form :~start,length, e.g. %N:~2,5 or %N:~8,5 . To append a number directly after the length value, use another ~ character, e.g. %N:~2,5~2. Negative values are measured from the end of the string. Example: %P:~0,-1 cuts off the backslash from the path.

The Start value -0 is special: %N:~0,20 copies the first 20 characters of the name without extension, %N:~0,-20 the first 20 characters of the extension without the name.

#### Special parameters:

**?** as the **first** parameter causes a **Dialog box** to be displayed before starting the program, containing the following parameters. You can change the parameters before starting the program. You can even prevent the program's execution.

**%P** causes the source path to be inserted into the command line, including a backslash (\) at the end.

**%N** places the filename under the cursor into the command line.

**%T** inserts the current target path. Especially useful for packers.

**%M** places the current filename in the target directory into the command line.

**%O** places the current filename **without extension** into the command line.

**%E** places the current **extension** (without leading period) into the command line.

**%B, %B0..%B9**

adds a directory name from the path (including the relative path from branch view, or the search result).

%B or %B0 = parent directory, %B1 = grandparent directory etc.

**%BT, %BT0..%BT9**

adds a directory name from the target path (without the relative path from branch view).

%BT or %BT0 = parent directory, %BT1 = grandparent directory etc.

**%B-, %B-0..%B-9**

adds a directory name from the path without the relative path from branch view, empty in

search result).

%B- or %B-0 = parent directory, %B-1 = grandparent directory etc.

### **%B+, %B+0..%B+9**

adds a directory name from the path (with branch view), counting from the start with the drive/server name:

%B+ drive letter with ":", %B+0 = drive letter without ":", %B+1 = first directory or share, %B+2 = second directory etc.

**%S** insert the names of all selected files into the command line. Names containing spaces will be surrounded by double quotes. Please note the maximum command line length of 32767 characters.

### **%S10**

insert the names of the first 10 selected files (max.) into the command line. Allows you to limit the number of file names passed to the program. You can use any other number.

### **%P%S**

insert the names of all selected files into the command line, with full path. Names containing spaces will be surrounded by double quotes. Do NOT put quotes around %P% S yourself!

**%R** like %S, but with selected names from the target panel

**%C1** Like the first parameter of "Compare by content": First selected file, or file under cursor

**%C2** Like the second parameter of "Compare by content": Second selected file, or first selected in target panel, or file with same name in target panel. Note: If the right panel is active and less than 2 files are selected, %C1 and %C2 will be reversed.

### **%C3..%C9**

Selected files Nr. 3 .. 9 in source panel, empty if not enough selected

### **%c1..%c9**

Like %C1..%C9, but with 8.3 names and paths

**Notes:** %N and %M insert the long name, while %n and %m insert the DOS alias name (8.3). %P and %T insert the long path name, and %p and %t the short path name (Same for %o, %e, %b and %s). %p/%t inserts the ftp URL of the directory for ftp connections. By putting %P, %p, %T or %t directly in front of %S or %s, the path name is inserted with the file name for each file. Example: %P%S inserts the long path and file name for all selected files.

%% inserts the percent-sign.

### **%L, %l, %F, %f, %D, %d, %WL, %WF, %UL, %UF**

create a list file in the TEMP directory with the names of the selected files and directories, and appends the name of the list file to the command line. The list is deleted automatically when the called program quits. Only one list per command is supported. Include parameter **%a** to also append selected files and directories from the target panel. 10 types of list files can be created:

**%L** Long file names including the complete path, e.g. c:\Program Files\Long name.exe

**%l** (lowercase L) Short file names including the complete path, e.g. C:\PROGRA~1\LONGNA~1.EXE

**%F** Long file names without path, e.g. Long name.exe

**%f** Short file names without path, e.g. LONGNA~1.EXE

**%D** Short file names including the complete path, but using the DOS character set for accents.

**%d** Short file names without path, but using the DOS character set for accents.

**%Q** Turn off automatic quotation marks around certain parameters like %P%N when the name contains a space. The user will then have to place them by himself.

### **%UL,%UF**

like %L and %F, but with UTF-8 Unicode list file (with byte order mark)

### **%WL,%WF**

like %L and %F, but with a UTF-16 Unicode list file (with byte order mark)

**%v** Insert virtual file name in case of file system plugins like "virtual panel", where %N pastes the name of the real file (in the file system) to which the entry points to

**%V** Like %v, but including the full path (including the plugin name)

**%X** Interprets the following parameters after this parameter as left/right instead of source/target:

%P, %p (left path), %T, %t (right path), %N, %n (left name), %M, %m (right name), %S, %s (left selected), %R, %r (right selected)

**Example:** %X%P %T sends left and right path to e.g. an external sync tool

- %x** Interprets the following parameters after this parameter again as source/target  
**Example:** %X%P %x%P sends left and source path to the called program
- %Y** anywhere in the parameters: Pass empty list to program when nothing is selected when using one of the List parameters like %L, or multi-file parameters like %S. Otherwise, the file under the cursor would be passed.
- %y** anywhere in the parameters: When nothing is selected when using one of the List parameters like %L, pass all other parameters except for the list parameter to the called program.  
**Note:** %Y%y works like %y when something is selected or the cursor stands on "..", otherwise it selects the file under the cursor.
- %Z** anywhere in the parameters: Allow to pass archives as path to programs for %P or %T when inside an archive  
**Example:** %Z%P passes name of archive to external tool when TC shows contents of an archive
- %|envvar|**  
 Insert environment variable **envvar** in the parameters field. Using %envvar% directly isn't possible, because the % sign is already used for placeholders. Therefore this modified syntax needs to be used. The "command" field still uses the normal %envvar% field.
- Only when defining alias commands:**
- %A** Inserts the rest of the entered command line
- %A1..%A9**  
 Inserts the first until the ninth parameter.  
 Example: Alias op points to Command: totalcmd.exe Parameter: /L=%A1 /R=%A2  
 -> Command line: op c:\dir1 d:\dir2 will create command:  
 totalcmd.exe /L=c:\dir1 /R=d:\dir2

**Run minimized:** Program will be started as an icon

**Show as menu:** Only for .bar files: Show linked button bar file as a popup menu instead of switching to the new bar. Use Shift to use the other option, e.g. show bar as menu if this option isn't checked.

**Run maximized:** Program will be started full screen

**Icon file:** File which contains the icon for the button bar. For Windows programs, you can specify the name of the exe file itself. (e.g. notepad.exe). For internal commands, there are some icons available in **wcmicons.dll**. If an alternative icon library is selected via [Configuration - Options - Icons](#), when you specify wcmicons.dll here (without path!), Total Commander will automatically substitute the list of icons from the configured library file.  
 The default icon for an extension .ext (e.g. for Windows store apps) can be specified in the form @anyname.ext

**Icon:** In this list, all icons contained in the specified **icon file** are shown. The number of the selected icon is shown in front of the list. Just click on an icon to select it for the button bar.

**Tooltip:** Content of the small window which is shown when the mouse cursor is pointed on a button for more than one second. A vertical line | will be shown as a manual line break. Two vertical lines || will be shown as one. A single space causes no tooltip to be displayed for that button.

**OK** Saves the changes made to the specified bar file.

**Cancel** Closes the dialog box without saving.

**Help** Opens Windows Help with this page.