

# CHANGE

User Manual

Version 2.05

John W. Clinton  
P.O. Box 678  
Boston, MA 02136  
Bix: JClinton  
CIS: 70244,1340

## **Introduction**

Please pass this software around to your friends and associates. Feel free to use this software for a month. If you find CHANGE as useful as I do, please Register!. Full registration is only \$25.00 in U.S. funds.

Registered Users receive the latest advertisement-free version of the software. Delivery options include EMail via Bix, CompuServe, or U.S. mail. Registered Users will also receive all updates made to CHANGE for a period of 1 year from the date of registration. Updates to CHANGE have been averaging one every six to eight months. Registered versions of the software are smaller and run faster. They do not have any of the advertising or other annoying banners. All of the output may be redirected using DOS redirection operators or may be suppressed completely. I often include a number of other useful utilities with the distribution media as space and availability permit. Any special modifications you may require are available to registered users by special arrangement. As a matter of policy I do not currently make the source for any products available.

### **Help support the Shareware Concept!**

## **Disclaimer:**

This software is provided 'as is' and is without any warranty of any kind, whether express or implied, including, but not limited to the implied warranties of fitness for a particular purpose. I will not be liable for any special, indirect, incidental, consequential or similar damages due to any loss of any kind related to the use of this software or for any other reason, even if I or an agent of mine has been advised of the possibility of such damages. In no event shall my liability for any damages ever exceed the price paid for the license to use software, regardless of the form of the claim. The person choosing to use the software assumes any and all risks as to the quality and performance of the software.

## **Acknowledgments:**

Special thanks to my wife Loretta for putting up with all of the time I put in on these little projects.  
Special thanks to Roedy Green for his constructive feedback on the program and documentation.

## CHANGE v2.05

### **Description:**

CHANGE allows you to make global changes to the contents of files. It works in much the same way as your favorite word processor's global change feature.

There are two major differences.

- The change(s) are propagated through all of the files in a set of comma delimited wildcard filespecs on the command line. This can include all files in all sub directories of the current sub directory or every directory on all disks attached to the system.
- You may specify all of the changes to be made with one pass through the file(s).

### **This utility is indispensable when:**

- Working with a large number of source files and you need to change variable or function names, literals, etc. globally.
- Working on old code and you want to retrofit it with new naming standards to bring the code up to date.
- Anytime there are a large number of files and a large number of changes that need to be made *quickly*.

### **Usage:**

If no parameters are passed in to CHANGE a usage summary will be printed. Additional help is available by using the /HELP switch.

### **In general:**

```
CHANGE <filespeclist> <matchString> <replacementString> [...] [/switches]
```

CHANGE - locates all occurrences of the specified file(s) and replaces the search string(s) with the specified replacement string(s).

Each filespec in the <filespeclist> may contain a drive, path and filename. The drive is the volume that will be searched. All files matching the files in the <filespeclist> will be search and modified as needed.

## CHANGE v2.05

### Examples:

Change all C sources in the current directory. Make 'aa' into 'bb'.

```
CHANGE *.c 'aa' 'bb'
```

Change all C source files in \SRC, all Headers in \INC and all files in \TMP. Make 'foo' into 'foobar'.

```
CHANGE SRC\*.C,INC\*.H,TMP\*.* 'foo','foobar'
```

Search and Replace strings are single words or quoted strings. Either single or double quotes will do. Everything on the command line is uppercased unless surrounded by quotes. The Search and Replace parameters may be separated by spaces or commas. If DOS redirection characters are part of any string on the command line they MUST be surrounded by DOUBLE QUOTES! That is the ONLY way to prevent DOS from usurping them.

### Customizing CHANGE

CHANGE is now being shipped with a utility called CHGINST. This program allows you to re configure the defaults of the most commonly used (but not all) switches to suit your own specific needs. To use CHGINST just run it from the command line. It will search for CHANGE in the current directory and along the DOS PATH. If change cannot be found it will stop and suggest the following:

Either move CHANGE into the same directory with CHGINST, move CHANGE into a directory along your DOS PATH, or invoke CHGINST with the name of the directory that CHANGE is in as a parameter.

### Example:

```
CHANGE is in C:\UTILS and CHGINST is in C:\ use  
CHGINST      -> if C:\UTILS is in your PATH,  
CHGINST C:\UTILS  -> if C:\UTILS is NOT in your PATH.
```

### Note:

CHGINST DOES depend on CHANGE being called CHANGE.EXE!

### General command line handling:

Switches are placed anywhere on the command line except within quoted strings. Switches may be abbreviated to the smallest unique string among the switches the program supports. The absolute minimum number of characters required for each of the switches is indicated by the uppercase part of the switch names on the help screen. Everything on the command line is converted to uppercase unless surrounded by quotes. Both double (") and single (') quotes are supported. If you need imbedded special characters or spaces in a parameter surround it with quotes. Text containing single quotes may be surrounded by double quotes, and text containing double quotes may be surrounded by single quotes. Non printable characters may be placed into the quoted strings on the command line or in indirect files (discussed below) using the following syntax: #nnn or ^a where 'nnn' is the Decimal ASCII code for the desired character and 'a' is '@'..'Z' corresponding to ^@ through ^Z.

## CHANGE v2.05

### DOS Notes:

1. DOS has a gotcha for strings like "Contains 4 spaces ' ' ". It will convert the 4 spaces to a single space before CHANGE gets control. This is only a problem for parameters entered on the command line. Indirect files handle this situation correctly. CHANGE also supports an environment string 'CMDLINE=' if it is present as a source for its command line. This permits 4DOS users to automatically avoid this problem. It also allows them to support up to 255 characters on the command line.
2. If your string(s) contain the DOS re-direction operators '<' '>' or '|' then you MUST enclose the string in DOUBLE QUOTES! Double Quotes will prevent DOS from using them for redirection of the standard input/output files.

### Examples:

#007 - Is the ASCII BEL Character.

^G - also the ASCII BEL character.

5#0075 - would be converted to 5's surrounding a BEL.

x#7x - would be converted to x's surrounding a BEL.

The leading zeros are required in ex. 1&3 to keep the other numbers from confusing CHANGE. They may be omitted when there is only one way to interpret the data. These special characters ( '#' and '^' ) may appear in your data as long as they are unambiguous. When in doubt, use two of the special characters to get one. See the examples below.

Normally the DOS command line supports parameters up to 128 bytes. Indirect parameter files may be used for changes that either require more than 128 bytes of input OR for CHANGEs that may be used again in the future. The total size of the parameters may now total in excess of 64KB.

## **Indirect Parameter Files:**

To use an indirect file place an '@' at the beginning of the name of the indirect file on the command line. The indirect file itself may contain anything that is normally allowed on the command line for CHANGE.

Switches specified in the indirect file may be overridden or added to by specifying the new switches on the command line following the indirect file's name.

```
CHANGE TEST.PAS @varchang.lis /SUBDIR
```

This would include the list of change strings found in the file VARCHANG.LIS (in the current directory). The lines in the file are concatenated together with a space appended to the end of each line. If the resulting command line has an odd number of CHANGE strings specified an error will be reported.

## **A reminder about Case:**

Remember that the command line is always uppercased first. If you are using the /CASE switch then you will probably want to surround your match and replace strings with quotes. If you want lower case in your replacements then be sure to surround the replace string with quotes! Be sure when using foreign language character sets or plain binary data that you use both quoting and the /CASE switch to prevent any potential problems with the interpretation of these characters!

## More Examples:

1. To rename the ever popular 'foobar' variable to 'MyVar' in the program 'test.pas' you would use the following CHANGE command:

```
CHANGE test.pas 'foobar' 'MyVar'
```

2. To change foobar to @#^foobar (use of special chars) double up the special chars. Use:

```
CHANGE *.pas 'foobar' '@@##^^foobar'
```

3. To change strings that contain special characters you need to represent the special characters in some fashion and 'allow' CHANGE to see them. Any character in the ASCII set may be specified in a QUOTED STRING by using a # followed by the character's three digit decimal ASCII code, for instance #007 is the BEL character.

To reformat a LF delimited text file into a more standard CRLF delimited file use the following:

```
CHANGE STRANGE.C ^J ^M^J
```

4. Several CHANGEs may be combined. The following CHANGE turns the variables called 'var1' into 'MyOtherVar' and reformats a LF delimited text file to a CRLF delimited text file.

```
CHANGE *.PAS 'foobar','MyVar' 'var1','MyOtherVar' #10,#13#10
```

5. If you have a large or complex change like the one above it is often better to use an indirect file. The indirect file, 'BIGCHG.IND', for that change would look like this:

```
!===Start of file BIGCHG.IND===  
! Comments are allowed within indirect files.  
! They are lines that begin with an '!'  
'foobar',      'MyVar'  
'var1',        'MyOtherVar'  
#10,           #13#10  
!===End of file BIGCHG.IND===
```

And the invocation of CHANGE would be:

```
CHANGE *.PAS @BIGCHG.IND
```

## **Change History**

A complete history of CHANGE has been moved to the file HISTORY.DOC.

## **Known Problems:**

The following problems are known to exist in this version:

1. When CHANGE is used on a file that had DOS file protection bits set, the protection bits are LOST!  
When CHANGE finds one of these files and it is told it is Ok to change it, CHANGE sets the file's protection to ARCHIVE only. When the original file is renamed to the .BAK version it is left with just the ARCHIVE bit set. The new version of the file is set to ARCHIVE only. It is left up to the User to reset the file protection appropriately.

*I don't view this as a bug, but you should be aware of it.*

2. When running CHANGE on a Netware file volume you must have the privileges required to edit the files and to create a subdirectory in the directory containing the file(s) to be CHANGED.

## **Planned Changes:**

CHANGE is currently 92% Turbo Pascal v6.0, the remainder is TASM.

If I get enough requests I will be adding unix grep style 'regular expression' pattern matching. If you need this please let me know! ( So far there has been 1 inquiry )

## **Encountering Problems:**

If you wish to report a problem with this software I can be reached at the above address, or send electronic mail to me via Bix (JClinton) or CompuServe (70244,1340). Please provide as much detail into the nature of the problem encountered, along with the system configuration details and SMALL examples of the data that demonstrates the problem.

As always I welcome new ideas, comments and suggestions!

## Appendix A

### Currently supported switches:

*(Only the upper case characters are required.)*

/Help	- Displays this message.
/[NO]ALL	- Causes all fixed drives to be completely searched.
/[NO]SHow_context	- Causes the line that matched to be displayed prior to the change and prompts the user to confirm that specific change. Changes may be allowed (Press 'Y'); Disallowed (press 'N' or RETURN); Make all of the changes found from here to the end of this file, (press ^Z); Make no more changes to this file, but keep the changes made so far (press ESC); or abort the changes made to this file completely, (press '^C'). Press ? for additional help at the CHANGE prompt. The default is /SHow_context in CHANGE as shipped, See the notes on the CHGINST program below.
/[NO]HEX_display	- When /SHow_context is in use this makes the HexViewer provide the first look at the file. 'A' and 'H' can still be used to switch between the Hex and ASCII viewers.
/[NO]Prefix_change	- Only make the CHANGE if the matchstring is at the beginning of a string. Match 'thisx', not 'xthisx' or 'xthis'.
/[NO]SUFFIX_change	- Only make the CHANGE if the matchstring is at the end of a string. Match 'xthis', not 'xthisx' or 'thisx'.
/[NO]Whole_words	- Only make a CHANGE if the matchstring is a whole surrounded by non alphanumerics. Match 'this', not 'xthis' or 'thisx'.
/[NO]Sub_directories	- Causes all sub directories of the current one to be searched. The default is \NOSubdirectories in CHANGE as shipped, See the notes on the CHGINST program below.
/[NO]Case_sensitive	- Allows the searching to be performed respecting the case of the match and replacement strings. The default is /NOCASE_SENSITIVE in CHANGE as shipped, See the notes on the CHGINST program below.
/[NO]Unique_bak	- Guarantees that backup files created will be unique. This will prevent CHANGE from deleting a backup file that already exists. CHANGE automatically enables this option when wildcard filespecs (like TEST.*) are specified that can overwrite backup files that CHANGE just created. The default is /Unique_bak in CHANGE as shipped, See the notes on the CHGINST program below.
/[NO]Floppysize=n	- Defines the size in KB of the largest floppy on your system you want searched. By default drives a: and B: are not checked at all. EX: /ALL/FLOPPYSIZE=1200 would check the size of each drive before searching. Only those 1.2MB and up would be searched.
/[NO]CONFIRM_change	- Prompts for permission to make changes to each FILESPEC matched that actually contains a match. This allows the user to selectively modify files.
/ANy_attribute	- Change all files matching the FILESPEC, regardless of special file attributes. /CONFIRM_CHANGE option is set by default, but may be overridden.
/[NO]Log	- Do [NOT] write progress messages to the console. The default is /NOLog in CHANGE as shipped, See the notes on the CHGINST program below.
/[NO]BACKups	- Do [NOT] rename the original files to .BAK. The default is /BACKups in CHANGE as shipped, See the notes on the CHGINST program below.
*/[NO]Quote_mode=?	- Do [NOT] use the quote 'respect' features. ?: Only_in_quotes == Just change things WITHIN quotes. Protect_quotes == Just change things OUTSIDE quotes. Ignore_quotes == (delivered default) change in and out of quotes. The default is /Quote_mode=Ignore

## CHANGE v2.05

\* /Q\_Uote\_char=?

- Specify a quote character.

?: Single\_quotes == Specify that quotes are "".

Double\_quotes == Specify that quotes are "".

<any\_char> == Specify a custom quote character.

This character may be (paradoxically) presented in quotes!

The default is /QUOTE\_CHAR='