

File Attribute Changer

Quick Start Guide

END-USER LICENSE AGREEMENT FOR FILE ATTRIBUTE CHANGER FREWARE

This End-User License Agreement ("EULA") is a legal agreement between you (either an individual or a single entity) and Hoai Technology, Inc. and describes your rights and obligations regarding the use of the File Attribute Changer Freeware software ("SOFTWARE").

If you do not agree to all of the terms of this agreement, you shall remove all the files related to the SOFTWARE from your storage media and cease any further use of the SOFTWARE. Any violation of these terms will subject you to legal liability.

The SOFTWARE is the intellectual property of and is owned by Hoai Technology, Inc.

The SOFTWARE is protected by copyright laws and international copyright treaties, as well as other intellectual property laws and treaties.

Hoai Technology, Inc. grants you the non-exclusive license to use the SOFTWARE for free. Such use is granted for personal, educational, and any other commercial or non-commercial purposes.

You may:

- a) Install and use the SOFTWARE on as many computers as you wish
- b) Modify or alter the SOFTWARE at your own risk
- c) Re-distribute the SOFTWARE provided that this license agreement is included with the SOFTWARE

You may not:

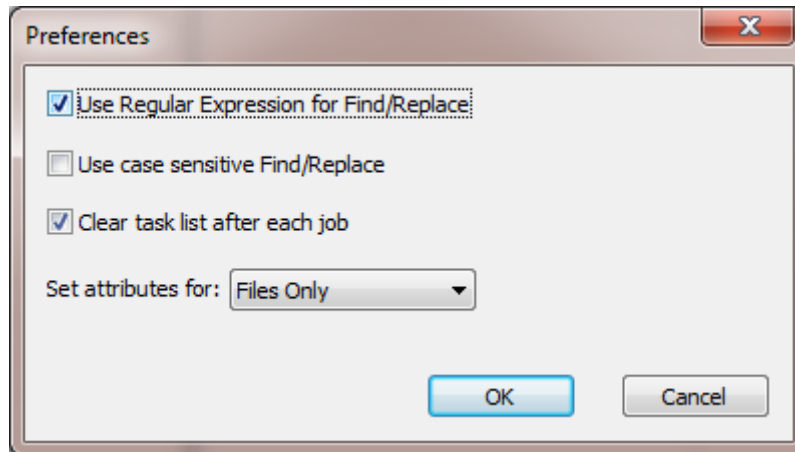
- a) Sell, rent, lease, or sublicense the SOFTWARE
- b) Remove any proprietary notices or labels on the SOFTWARE
- c) Use the SOFTWARE as a basis for creating products that provide the same, or substantially the same, functionality as the SOFTWARE.

This EULA has no termination date. Hoai Technology, Inc. may terminate this license agreement if you fail to comply with the terms and conditions of this EULA. In such an event, you shall remove all the files related to the SOFTWARE from your storage media and cease any further use of the SOFTWARE.

This SOFTWARE is provided "AS IS" without warranty of any kind including, but not limited to, warranties of merchantability, fitness for a particular purpose and non-infringement. In no event will Hoai Technology, Inc. be liable for any direct, indirect, incidental, special, exemplary or consequential damages, including damages for loss of profits, loss or inaccuracy of data, incurred by any person from such person's usage of this SOFTWARE even if advised of the possibility of such damages.

1. Regular Expression Basic

File Attribute Changer's basic search and replace is simple and straight forward. However, the basic search feature is limited in term of fine tuning the search criteria. To enable the advance search feature, we can turn on the regular expression search function in the preference dialog.



File Attribute Changer supports Perl regular expression syntax. POSIX basic syntax and POSIX extended syntax is disabled, and therefore, not supported. In Perl regular expression, all characters match themselves except for the following special characters:

. [] { } () \ * + ? | ^ \$

1.1 '.' Dot Character

The single character '.' when used outside of a character set will match any single character.

1.2 [] Square Brackets

Match anything inside the square brackets for ONE character position once and only once. For example, [12] means match the target to 1 and if that does not match then match the target to 2 while [0123456789] means match to any character in the range 0 to 9.

1.3 { } Curly Brackets or Braces

{n} Matches when the preceding character, or character range, occurs n times exactly, for example, to find a local phone number we could use [0-9]{5} which would find any number of the form 12345.

{n,m} Matches when the preceding character occurs at least n times but not more than m times, for example, 'ca{2,3}b' will find 'caab' and 'caaab' but NOT 'cab' or 'caaaab'.

{n,} Matches when the preceding character occurs at least n times, for example, 'ca{2,}b' will find 'caab', 'caaab' or 'caaaab' but NOT 'cab'.

1.4 () Parentheses

Used to group (or bind) parts of our search expression together. Officially this is called a subexpression (a.k.a. a submatch or group) and subexpressions may be nested to any depth. Parentheses (subexpresions) also capture the matched element into a variable that

may be used as a backreference

1.5 \ Backslash

When follow by a special character, it is used to indicate that the special character should be treated as a literal (search character) not a special character. Can also be used with other character to perform certain functions

1.6 * Asterisk

The * asterisk character is used to match the preceding character zero or more times. For example, the expression `zo*` matches either `z` or `zoo`.

1.7 + Plus

The + plus character is used to match the preceding character zero or more times. For example, the expression `zo+` matches `zoo`, but not `z`.

1.8 ? Question Mark

The ? question mark character is used to match the preceding character zero or one time. For example, the expression `labou?r` matches both `labor` and `labour`.

1.9 | Vertical Bar

Find the left OR right values, that is `x|y` matches either `x` or `y`. For example, `foo|bar` matches `foo` or `bar`.

1.10 ^ Caret

Look only at the beginning of a line or string.

1.11 \$ Caret

Look only at the end of a line or string.

These are some of the most basic syntax of regular expression. There are many other special characters and functions that regular expression can use. Please consult the regular expression manual or search online for additional information.

2. Examples

2.1 Wildcard Search

To perform a wildcard search, we can combine the `.` dot character and the `+` plus character. This will match any string containing text, but will not match an empty string. We can use `^.*$` to search for both an empty string and a string containing text. We can also use `^$` to match only the string that is empty.

File Name	Current File Name	New File Name
Find what: .+	Name	Name
Replace with: New File	[ENG] My old file.pdf	New File
<input type="checkbox"/> Use numbers to replace #'s	[FRN] My old file.pdf	New File
Start At: 0	My old file part 1.jpg	New File
	My old file part 2.jpg	New File
	My old file part 3.jpg	New File

2.2 Backreference

We can use the () parentheses to group the search term, and use that for backreference in the replace term.

File Name	Current File Name	New File Name
Find what: .+(\.+.+)	Name	Name
Replace with: Test\1	[ENG] My old file.pdf	Test.pdf
<input type="checkbox"/> Use numbers to replace #'s	[FRN] My old file.pdf	Test.pdf
Start At: 0	My old file part 1.jpg	Test.jpg
	My old file part 2.jpg	Test.jpg
	My old file part 3.jpg	Test.jpg

From the above example, the expression `.+` search for the file name, while the expression `(\..+)` is used to search for the file extensions, and it also allows us to use backreference `\1` in the replace expression. Here's another example:

File Name	Current File Name	New File Name
Find what: (\[.+\]) (My.+)\.	Name	Name
Replace with: \2 \1.	[ENG] My old file.pdf	My old file [ENG].pdf
<input type="checkbox"/> Use numbers to replace #'s	[FRN] My old file.pdf	My old file [FRN].pdf
Start At: 0	My old file part 1.jpg	My old file part 1.jpg
	My old file part 2.jpg	My old file part 2.jpg
	My old file part 3.jpg	My old file part 3.jpg

Here, the first part of the search expression `(\[.+\])` search for anything inside the [] square bracket in the file name, while the second part `(My.+)` search for the rest of the file name whose name begin with My. The last part of the search expression `\.` is used to find the period. If we want to be more general, we can use the search expression `(\[.+\])(.+)\.` which uses the wildcard search instead of the more specific file name.

2.3 Change File Name Case

There are many reasons we want to change the case of a file name. Regular expression can easily accomplish this task. For example, to change the file name to all upper case, we can use the search expression **(.+)** and the replace expression **\U\1\E** as shown below.

File Name	Current File Name	New File Name
Find what: (.+)	Name My zip file part 1.zip	Name MY ZIP FILE PART 1.ZIP
Replace with: \U\1\E	My zip file part 2.zip	MY ZIP FILE PART 2.ZIP
<input type="checkbox"/> Use numbers to replace #'s	My zip file part 3.zip	MY ZIP FILE PART 3.ZIP
Start At: 0		

To change the file name to lower case, we use the search expression **(.+)** and replace express **\L\1\E** as shown below.

File Name	Current File Name	New File Name
Find what: (.+)	Name My zip file part 1.zip	Name my zip file part 1.zip
Replace with: \L\1\E	My zip file part 2.zip	my zip file part 2.zip
<input type="checkbox"/> Use numbers to replace #'s	My zip file part 3.zip	my zip file part 3.zip
Start At: 0		

We can also change the file name to title case by using the search expression **\b([a-z])** and the replace expression **\U\1\E** as shown below.

File Name	Current File Name	New File Name
Find what: \b([a-z])	Name My zip file part 1.zip	Name My Zip File Part 1.Zip
Replace with: \U\1\E	My zip file part 2.zip	My Zip File Part 2.Zip
<input type="checkbox"/> Use numbers to replace #'s	My zip file part 3.zip	My Zip File Part 3.Zip
Start At: 0		

Changing the file name to sentence case is also easy. Use the search expression **(^[a-z])(.+)** and the replace expression **\u\1\L\2\E** to accomplish this task.

File Name	Current File Name	New File Name
Find what: (^[a-z])(. +)	Name MY ZIP FILE PART 1.ZIP	Name My zip file part 1.zip
Replace with: \\u\1\\2\E	MY ZIP FILE PART 2.ZIP	My zip file part 2.zip
<input type="checkbox"/> Use numbers to replace #'s	MY ZIP FILE PART 3.ZIP	My zip file part 3.zip
Start At: 0		

2.4 Auto Numbering

To use the auto numbering feature, select the “Use numbers to replace #'s” checkbox in the File Name tab.

File Name	Current File Name	New File Name
Find what: part [0123456789]	Name [ENG] My old file.pdf	Name My old file [ENG].pdf
Replace with: ### of 3	[FRN] My old file.pdf	My old file [FRN].pdf
<input checked="" type="checkbox"/> Use numbers to replace #'s	My old file part 1.jpg	My old file 0001 of 3.jpg
Start At: 1	My old file part 2.jpg	My old file 0002 of 3.jpg
	My old file part 3.jpg	My old file 0003 of 3.jpg

3. Helpful Tips

- To edit a task item, double click on it.
- Double click on the file name will copy the name/date/attribute to the find/replace window.
- When include sub-directory is selected, double click on the folder will browse to that folder. (Only supported on Windows Vista and later)
- Click on the file name field and type Ctrl+C will copy the file name to the clip board. The same can be done for the date field
- Right-click on the header of the file listbox to select other hidden columns
- File Attribute Changer can save the config.ini file in the AppData folder. To do this, create a file name AppData-config.ini in the same location as the File Attribute Changer program.
- Start File Attribute Changer from the command line. File Attribute Changer support the following arguments:
 - -FilePath Specify a file or folder to open.
 - -TaskList Specify a task list to open.
 - -Apply Execute the tasks. Require both the FilePath and the TaskList.
 - -NoConfirm Apply without display a confirmation dialog.
 - -? Help menu.
- Example usage:
 - File Attribute Changer.exe [-FilePath] [-TaskList] [-Apply] [-NoConfirm] [?]

- File Attribute Changer.exe D:\MyPicture.jpg
- File Attribute Changer.exe -FilePath D:\MyPicture.jpg
- File Attribute Changer.exe -TaskList D:\tasklist.fac
- File Attribute Changer.exe "D:\My Vacation Picture.jpg" D:\tasklist.fac
- File Attribute Changer.exe "D:\Picture Folder" "D:\Rename tasklist.fac" -Apply