

## types of filter files

Definition. PowerReplace recognizes automatically two types of filter file: if a filter can be represented by a conversion table ( or a dictionary) like:

```
"\d000"    "seconf-stuff of 000"  
"\d001"    "seconf-stuff of 001"  
.....  
"\d255"    "seconf-stuff of 255"
```

this filter is called of type I. All other filters are called of type II.

In particular, the first-stuff of each line in the type I filter represents only one character.

Example 1. A filter of type I:

```
"á"        "&aacute;"    % first-stuff = "á", one characters !  
"\d65"     "a"          % first-stuff = "A", one characters !  
"B"        "b"          % first-stuff = "B", one characters !  
"\n"       "LF"         % first-stuff = LF, one characters !
```

Example 2. A filter of type II:

```
"á"        "&aacute;"  
"AB"       "C"          % first-stuff = "AB", two characters !  
"\n"       ""
```

Example 3. A filter of type II:

```
"A"        "B"  
"A"        "C"
```

PowerReplace can not construct a conversion table because "A" is defined twice.

Example 4. A filter of type II:

```
"A"        "B"  
"B"        "C"
```

If we construct a conversion table, it will be not equivalent to the filter. For example, this filter changes the text "ABC" to "CCC". But if we construct a conversion table, we obtaine an other output text "BCC".

Set filter type. To set the type of a filter, put the following define line at the beginning of the filter:

```
#type 1
```

or

```
#type 2
```

See the filter "ShowASCII" in the Filter folder for an example.

Profit.

- If your mac has enough memory (RAM) and your filter file has only a few lines, PowerReplace runs faster for type II filter than for type I.

- else, PowerReplace works better for type I filter: it needs only a little memory, and runs very fast.

### Speed Comparison

#### Configuration:

- Macintosh: test on PowerMac 6100/60 with system 7.5.5.
- Input text file: repeat the string "abcdefáSDà12âFDéjiè=+ê--úRFDù<<ûEDülmk <CR>" until file size >= 1500 KB.

Application	Operation	Time
-------------	-----------	------

### ptimizing Filter for Speed: An Example.

We need change a 8 bits text to html format. We can write a filter file as (Filter E, 59 lines):

% first, change accented letters to html format.

```
"á"      "&aacute;"
"à"      "&agrave;"
```

.....

```
"û"      "&ucirc;"
"ü"      "&uuml;"
```

% then change double-CR to paragraphe <P>,

% change simple-CR to <BR>

```
"\r\r"   "<P>"
"\r"     "<BR>"
```

Filter E is of type II. If your input text is very big, it is possible that the conversion is not fast enough. In this case, you can cut the filter E into two filters as following to optimize for speed:

Filter F (type I, 57 lines):

% first, change accented letters to html format.

```
"á"      "&aacute;"
"à"      "&agrave;"
```

```
.....
"ô"      "&ucirc;"
"ü"      "&uuml;"
```

And Filter G (type II, two lines):

```
% then change double-CR to paragraphe <P>,
% change simple-CR to <BR>
"\r\r"   "<P>"
"\r"     "<BR>"
```

Speed comparison (with the same configuration and input text as above)

Application	Operation	Time
-------------	-----------	------

## sterisk-sign(\*) in the First-String: An Example.

If the filter is:

```
"A*B"      "..."
```

and the input text is "XYAABBBAABBCD",

then the output text will be "XY...BB...B...BCD".

## Writing and Debugging Complicate Filter

PowerReplace can do more complicate text conversion. In the example in the “Tutorial Examples:Chapter 7:Write and Debug Filter”, we need select some information from the input file IllustratorText.

This input file is not a “good” text file. It has a good view in Illustrator, but not in text editor. We want to extract only the lines beginning with %AI3\_Note:, and some text after these lines. The following text is what we want extract from IllustratorText.

```
%AI3_Note:
(Black & White)
%AI3_Note:Ptext 13_Jahrom_47848
(Jahrom) Tx
%AI3_Note:Ptext 7_AI Kuwayt_29783
(AI Kuwayt) Tx
%AI3_Note:Ptext 1_Kuwait_67972
(Kuwait) Tx
%AI3_Note:Ptext 14_Taft_47045
(Taft) Tx
%AI3_Note:Ptext 1_Iraq_24837
(Iraq) Tx
```

Please see the complete input/output/filter files in the “Tutorial Examples:Chapter 7:Write and Debug Filter” folder.

◇ If you write a complicate filter, it is recommended that you use the “Convert In->Out” mode to debug your filter. This mode saves your times. See the section Debugging Filter in the Chapter 4: Tutorial.

◇ You can change the text style or text color in the Input and Output windows. Using ResEdit, change the resource 'Ttxt', ID=141 for Input and ID=142 Output.

## Length of Line and String

The length of each line must be less than 4096 characters. If the first-stuff is a single character in a line, then the second-stuff in this line must have less than 32 characters. The length of the string in the progress window must be less than 200 characters.

### Limitations:

- NULL character in input file

If your input file contains NULL characters (ascii = 0), PowerReplace may not work correctly. To solve this problem, you can save all NULL characters at the begin, and restore them at the end of your filter file. Your filter looks like:

```
"\d000"      "[NULL]"  % make sur no [NULL] in input file
.....
.....
.....
"[NULL]"     "\d000"
```

- Big Insertion

If you use insertion tag ">" in the second-string, then the first-string-found (the "source") must have less than 4096 characters.

Bad example: Your input text is of length 10KB, and looks like

```
Begin of file
Chapter 1: Introduction
.....
.....
.....
Chapter 9 : Conclusion
Ene of file
```

The following filter may cause error:

```
"Chapter * :."      "toto"
```

because between "Chapter 1" and ":", there is no space. If there is no ":" in the middle of your input text, then the first-string-found is:

```
Chapter 1: Introduction
.....
.....
.....
Chapter 2 :
```