

TEXTPIPE SUBFILTERS -

WHAT ARE THEY AND HOW DO I USE THEM?

- What is a subfilter? 1
- So what does this do? 2
- Now why would we do this? 2
- How do I create a subfilter? 2
- Preprocessing subfilters 2
- Search and replace filters as subfilters 2
- Comment filter 3
- Restriction subfilters 3
- Using VBScript/JScript to turn subfilters on and off..... 3
 - Processing text in one piece 3
 - Processing text as several pieces..... 4
- Database subfilter 5
- More White Papers and Documentation like This 5
- TextPipe Pro Evaluation..... 5
- Contact Details..... 5

What is a subfilter?

TextPipe subfilters are one of its most powerful features. Put simply, a subfilter is where one filter appears *inside* another, rather than simply beneath it. See below.

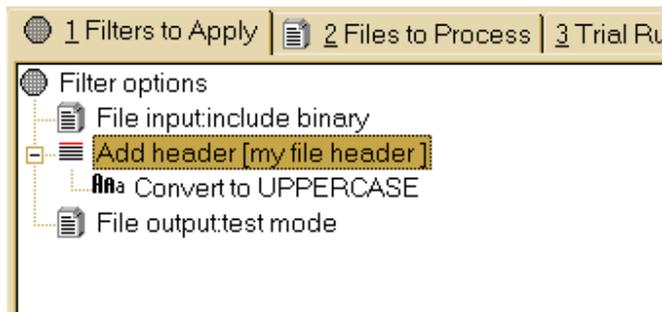


Figure 1. The UPPERCASE filter is a *subfilter* of the Add Header filter.

In the figure above, the *Convert to UPPERCASE* filter is a *subfilter* of the *Add header* filter. See how it is indented to the right? This is exactly how subfolders appear in Windows Explorer. The filter in the position of the *Add header* filter is often called the *parent*, while the filter in the position of the *Convert to UPPERCASE* filter is often called the *child*.

So what does this do?

In this example, the subfilter converts the text 'my file header' to 'MY FILE HEADER', **BEFORE** it gets used for the file header. The subfilter here performs some pre-processing on the actual data we've entered.

Now why would we do this?

While we could have typed in the text 'MY FILE HEADER' in the first place, if the header data was coming from an external file, or there was a lot of it, or we needed to make many changes to it, a subfilter comes in very handy. Normally we would have to design another TextPipe filter to fix up this data before we used it, but we can save that effort here by using a subfilter.

Subfilters also have other uses - read on for more details.

How do I create a subfilter?

To create a subfilter, just drag and drop a filter onto the text of the parent filter (if you don't drop over the text, the filter will be dropped above or below, not as a subfilter).

Note:

1. Not all filters support subfilters. In this case the drag and drop will actually drop *below* the target filter.
2. If you use subfilters at the end of the filter list ie just above the Output filter, you will find that all new filters you add get added as subfilters, even if you select the Output filter first. You can get around this by dragging the filters on top of the Output filter.
3. You can make a filter into a subfilter of the filter above it by right clicking the child filter and selecting *Make into a subfilter*.

Preprocessing subfilters

Filters where the subfilters perform pre-processing include

- Add header
- Add footer
- Add left margin
- Add right margin
- Add column
- Remove lines from list
- Remove lines matching pattern list
- Retain lines from list
- Retain lines matching pattern list
- Add text side-by-side
- Add repeating text side-by-side

Search and replace filters as subfilters

The search and replace filter is the most powerful subfilter. Using the perl or EasyPattern pattern matching language, you can locate inexact sections of text and then apply any filters to them.

Take this example:

```
Perl pattern [alt=".*"] with [$0]
|
+--Convert to UPPERCASE
```

The subfilters apply to the REPLACEMENT text, which means that you can re-arrange the found text first before sending it to the subfilter. If you want to pass the found text UNCHANGED, specify a replacement text of '\$0' - this substitutes the entire found text.

You can use a perl pattern's non-capturing brackets to force a search/replace to only find text when specific text is found before or after the search text. Text from non-capturing brackets is not included in the replacement text, and this can be very handy. You can also force it to find text where specific text is NOT before or after the search text.

Comment filter

The comment filter is designed to have subfilters. It doesn't perform any processing, but it's a neat way to group a set of filters with a comment about their purpose.

Restriction subfilters

Restrictions are a special class of subfilters. They limit how much text is affected by an operation e.g.

```
|--Restrict lines:Line 1 .. line 1
|
|   |--Replace [old value] with [new value]
|       [ ] Match case
|       [ ] Whole words only
|       [ ] Case sensitive replace
|       [ ] Prompt on replace
|       [ ] Skip prompt if identical
|       [ ] First only
|       [ ] Extract matches
```

2 By the way, if you've never seen filters displayed as above, check out the File Menu\Export Filter to Clipboard option)

For an excellent tutorial on restrictions, please go to

www.datamystic.com/docs

You can find all the restriction filters in Filters Menu\Restrict Menu.

Using VBScript/JScript to turn subfilters on and off

Now this is really advanced. If you've read our document on the Scripting filter, you'll know that you can use it to achieve any custom processing you require. With the addition of subfilters, you can use the script to **turn on or off** the subfilters you've defined, according to programmed criteria, such as every odd or even line, or for 5 lines after a line matching the word 'header'.

You can use one of two methods - either use the script to pass an entire piece of text in one shot, or use the script to pass several pieces at different times, with you telling it the start and end.

Processing text in one piece

With this method you pass some text to the subfilter and it gives you the result straight back.

```
processLine = TextPipe.subFilterEntireText(line) & EOL
```

Processing text as several pieces

With this method you initialize the subfilters, pass multiple pieces of text to it, and then flush the output. The subfilter takes care of passing the data on to the next part of the filter.

```
dim lineNumber

function processLine(line, EOL)

    lineNumber = lineNumber + 1

    if (lineNumber = 100) then
        'start the special processing on line 100
        TextPipe.initSubFilters
        TextPipe.subFilterChunk( line & EOL )
        processLine = ""
    else if (lineNumber > 100) and (lineNumber < 200) then
        'special processing for lines 100..200
        TextPipe.subFilterChunk( line & EOL )
        processLine = ""
    else if (lineNumber = 200) then
        'end special processing on line 200
        TextPipe.flushSubfilters
        processLine = line & EOL
    else
        processLine = line & EOL
    end if

end function

function endFile()

    if (lineNumber = 100) then
        endFile = ""
    else if (lineNumber > 100) and (lineNumber < 200) then
        TextPipe.flushSubfilters
        endFile = ""
    else if (lineNumber = 200) then
        TextPipe.flushSubfilters
        endFile = ""
    else
        endFile = ""
    end if

end function

function startJob()
    lineNumber = 0
end function
```

Database subfilter

If you extract BLOB data from a database with the database filter, the subfilter, if present, is used to encode the binary data.

```
|--Database Provider=MSDASQL.1;Persist Security Info=False;Data
Source=Crystal
| | Timeout: 30 seconds
| | [X] Generate header
| | XML
| |
| |--MIME Encode (Base 64)
```

More White Papers and Documentation like This

Available from:

www.datamystic.com/docs

TextPipe Pro Evaluation

You can download a free 30 day trial of TextPipe Pro from

www.datamystic.com/textpipe-wp.exe

You can also access our other downloads from

www.datamystic.com/freetrials.html

Please contact us if you have any questions, difficulties or queries.

Contact Details



DataMystic

5 Bond Street

Mt Waverley

Victoria 3149

Australia

Web site: www.datamystic.com

Phone: +61-3 9913-0595

Fax: +61-3 8610-1234