

52

Manage Find/Replace settings

Overview

The [FM community](#) and also the [FM bug base](#) request an extension to the standard Find/Change mechanism of FrameMaker to be able to save and retrieve settings by a name [2020-09-22].

This script **FMfindRepl** fulfils this at least partially.

Contents

Introduction	2
Installation.....	2
Working with FMfindRepl	3
Main panel.....	3
Special find types.....	7
Find types not implemented	9
FMsearchReplace.xml	9
Handling Regular Expressions.....	10
Transfer to the FM F/C panel.....	14
Known issues.....	15

Introduction

Installation

- 1 Download the ZIP from the [website](#) and expand it to a location of your liking.
- 2 Execute the file `PrepareInstallation.exe` (run as Administrator)
- 3 The language of the installation dialogues adapts to the UI language of your current FM installation (de, en, fr).
- 4 This program modifies some values in the file `InstallThis\config.ini`.
- 5 All FM-versions present in your `%appdata%/Adobe/FrameMaker/` are listed in the dialogue (on your system there may be only one).
- 6 Use **Continue** to open the installation program.
- 7 Select the appropriate option(s):
 - Use the first option to start the script automatically at FM-start. In this case the script and related files are stored in `%appdata%/Adobe/FrameMaker/vv/Startup/` and in a subfolder `FMfindRepl`.
 - Use the second option if you wish to start the script on demand from a trusted source. You can define the script in `File > Scripts ... > Cataloge (Script Library)` as Favourite: use the **Add** button and navigate to `My Documents\Adobe Scripts\FMfindRepl\FMfindRepl.jsx`.
- 8 With **Install** the script and associated files will be transferred into the relevant locations.
- 9 A message indicates the end of the installation. The desktop icons will be removed.

Running the script

The script establishes an entry in the edit menu¹⁾ with short cuts:

Documents	Manage Find/Replace Settings	ESC q f d
Books	Manage Find/Replace Settings	ESC q f b

The script opens a panel which can react to the change of document.

An ancillary program (a compiled²⁾ AutoHotKey script) handles the transfer of data from the main panel to the FM F/C panel.

The script adapts its UI to the language of FrameMaker (de, en, fr).

Test file The [test file \(FM-12 mif\)](#) comprises many types of objects, including colour misuse and overflow text.

- 1 In FM < 16 the item can not be placed correctly in the vicinity of the original Find item, It hence appears at the end of the menu.
- 2 It is compiled only because You may not have an AHK interpreter on your system. You check in the script folder for the source code.

Working with FMfindRepl

Intended work flow

It is not possible to retrieve information from the FM Find/Change panel for storage and retrieval³⁾. Hence I need to resort to the old principle of “divide and conquer” and separate the two tasks Find and Replace:

- The user sets up the Find (and Replace) details in the script panel.
- The current settings can be saved with a name in the script panel.
- Later in time settings can be retrieved, probably modified and transferred to the FM F/C-panel for execution.
- **Change, Change and Find** or **Change all** can only be achieved by the FM Find/Change panel.
- If only a find operation is required, it mostly can be achieved by the script⁴⁾. If also a replace operation is required, the settings are transferred to the FM find/change panel and handled there.
- Some special find operations can be achieved only by the script. Most of these searches look into the whole document (all sorts or pages and flows).

Main panel

Find Type (A)

Catalogue entries (B)

Find what (C)

Options (D)

Button RegEx Help (E)

Replace mode (F)

Replace by (G)

Look in (H)

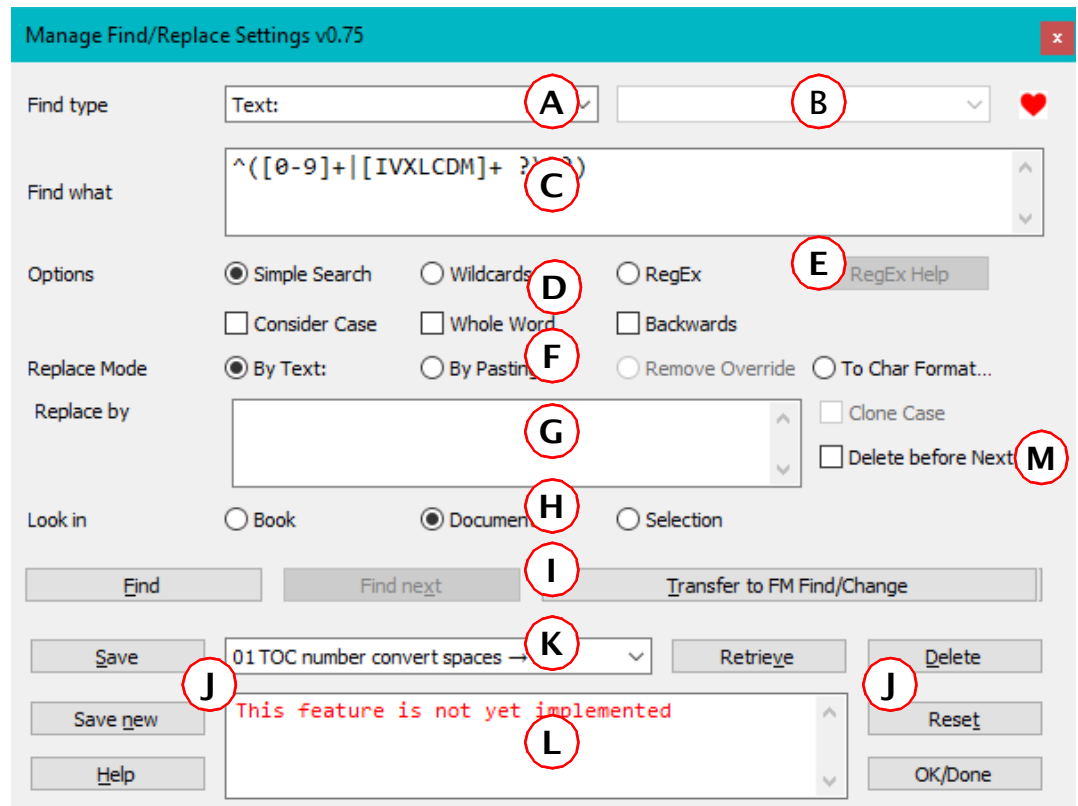
Action Buttons (I)

Maintain the settings (J)

List of saved settings (K)

Ancillary area (L)

Empty Text Line deletion (M)



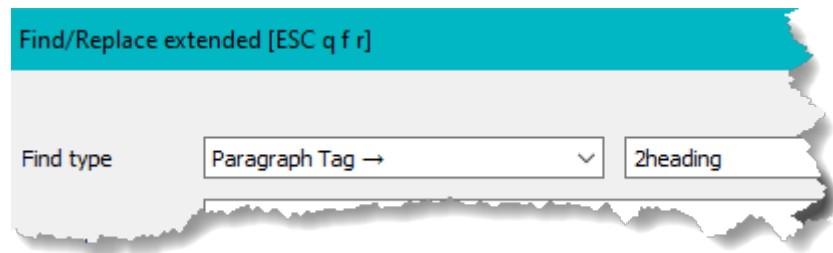
Find Type (A)

Drop down list of the types Anchored Frame ... Variable named → . Some of the entries have different labels to the original FM F/C panel. Some are left out and some are added.

- 3 Only edit fields can be ‘queried’.
- 4 See later for restrictions here.

Catalogue entries (B)

Find Types with an ending arrow (→) fill the adjacent drop down list with the names from the corresponding catalogue



This concerns the following types:

- Character Tag →
- Cross-Reference of Format →
- Object Style Tag →
- Paragraph Tag →
- Table Tag →
- Variable named →
Both user variables and system variables are listed in alphabetic order.

Special find types

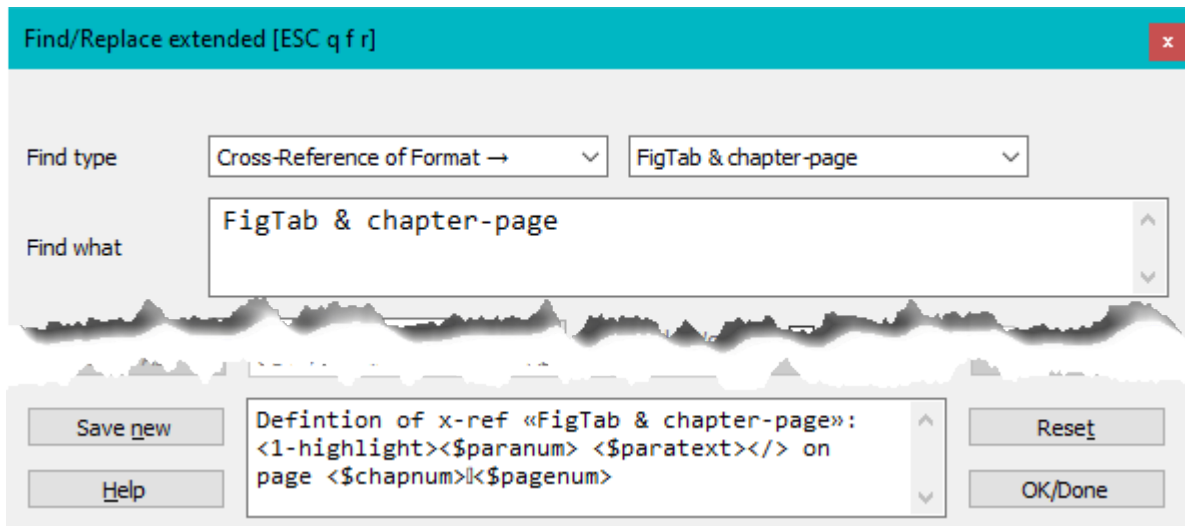
FMfindRepl offers some additional find types compared to the FM Find/Change dialogue. These can not be transferred for Find/Replace operations. See [Special find types](#) on page 7.

Find types not implemented

Due to lack of knowledge or information some find types present in the standard FM Find/Change panel are not implemented. See [Find types not implemented](#) on page 9.

Additional information

If a *Cross-Reference format* or a *Variable name* is selected, then the definition of this object is displayed in the info area. This lets you judge whether to look for the intended object:



Find what (C)

Text entry area for a find string or regular expression. To find uncatalogued tags/formats, enter the name here.

Options (D)

Radio buttons Simple Search, Wildcards and RegEx.

The contents of the **Find what** field is used as a starter in the panel where you create the RegEx.

The created RegEx will be inserted into the edit field **Find what**. Only elements for the Perl flavour of RegEx are imple-

mented. Hence you get a warning if your installation defines another flavour.

A simple test avoids the useless message “Not found” in case the RegEx is not valid.

See [Handling Regular Expressions](#) on page 10.

Whole Word	If the search string shall match only full words, check this. <i>Disabled for RegEx.</i>
Consider case	If case is relevant for the search, check this. May be overwritten by a RegEx.
Backwards	To search backwards from the current location, check this. <i>Disabled for RegEx.</i>

Button RegEx Help (E)

This becomes active only for the selection RegEx in the drop down list. It opens a panel from which Regular Expression elements can be selected.

Replace mode (F)

Replace methods are handled only in the FM F/C dialogue after transfer (See [Action Buttons \(I\)](#) on page 5).

- Radio buttons specifying the mode of replacement:
 - To Text
 - By Pasting
 - Remove Override
- To Character Format... This is not handled in the script and hence must be set up in the FM F/C panel.

Replace by (G)

The text to be used for replacement. This may be a regular expression if RegEx is selected in the Options. To remove found text, the replacement is empty.

Clone case Check, if case of found item shall be used for the replacement.⁵⁾

Look in (H)

Default is Document. **The other options will be available later in the development process:**

- If panel is opened while book is active, then Book is checked.
- Selection must be set by the user.

Note: *Search in selection is not implemented - you may wish to hand over to the FM F/C panel.*

Action Buttons (I)

The action buttons are active only if the proper settings are defined.

Find	Find the text or object according to the above settings.
Find next	Find the next occurrence of the object. F3 to repeat the find is reserved for the FM F/C panel.
Transfer to FM F/C	This hands over the settings to the FrameMaker Find/Change panel, where you can execute the Find, Change, Change & Find and Change All operations.

⁵ The detailed function of this is not clear to me.

Maintain the settings (J)

Save the current settings of Find/Replace with a name. Unfortunately ExtendScript does not allow to type anything into a drop-down list (there is no combo). To distinguish the two use cases (modify already saved settings, save new settings) two buttons are required.

Modify the current setting
(displayed name)

- ▶ Retrieve the appropriate settings.
- ▶ Modify the settings and check them.
- ▶ You may modify the info area; for example, add information about the modification.
- ▶ Use button **Save** to save the modified setting with the already defined name.

Create a new item in the settings

- ▶ If necessary, select all (**CTRL+a**) in the info area and **delete** it.
- ▶ Type the name for the new item into the info area: This name should be different to already defined names - although you may distinguish further by the additional information given.
- ▶ Use CTRL+Enter to add a line break. This is necessary to get an own paragraph in the info area.
- ▶ In the rest of the info area type the required text.

Exchange company name with paste buffer

Insert the current company name into the «Find What» field and have the replacement (variable) in the paste buffer.

- ▶ Use button **Save new** .

List of saved settings (K)

The top item of the list is empty. If you select something, button **Retrieve** becomes active.

Names *not* starting with a Latin character (e.g. by Cyrillic ones or \$, @ [...]) are sorted after the blank before the ones with Latin characters.

```
01 Remove manual list numbering
02 Exchange company name with paste buffer
Even this awful long name is displayed completely in the list
Test case: Ample gugus & other stuff
TOC number convert spaces → TAB
```

Thus you can easily prefer certain settings or group by prefix.

Retrieve settings

Button **Retrieve** reads the data of the selected item and sets the various options accordingly.

As a side effect the clipboard is filled with the data read in. This can be used, for example, for documentation purpose.

Example

Pasted clipboard is formatted with `lw-list-wide`:

Testing Write/Read keystroke file

Info Find Conditional Text.:These details cannot be stored in FMfindRepl - and must be set in the FM-palette.

Find type Conditional Text...

Find what This will not be used

Options Simple Search, Consider Case, Clone Case,

Delete item in list

Replace by IMHO also this will not be used

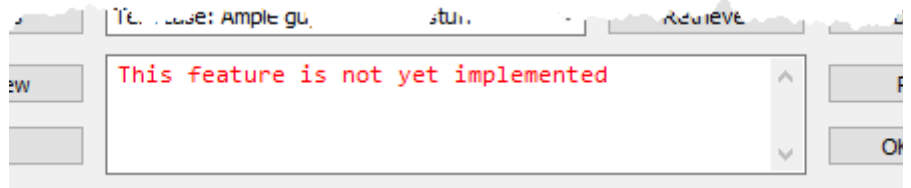
Use button **Delete** to remove the currently selected item from the list and the settings file. There is no undo for this.

Note: *Before you re-invoke the panel after the deletions you can manually recover by copying*

`%appdata%\Adobe\FrameMaker\vv\Startup\FMfindRepl\FMfindRepl_backup.xml to FMfindRep.xml.`

Ancillary area (L)

The ancillary area comprises additional buttons and the information area: This is used for displaying additional information from the retrieved settings; display properties for some Find types and it also may contain error messages.



Button Help

Open the PDF documenting this script.

Button Reset

Resets the panel to default settings.

OK/Done

Leave the panel. Same function as the top right **X** button.

Special find types

Rationale

Long living documents accumulate unwanted objects, such as empty TextLines or Overflowing Cells. For clean-out purpose it is also desirable to find unintended use of colour.

FMfindRepl offers some additional find types compared to the FM Find/Change dialogue. These can not be transferred for Find/Replace operations.

The scope of search for these functions may differ from that of the ordinary searches performed by FM:

Colour →

The catalogue lists both the user defined colours and the system standard colours - both in alphabetic name order.

Searching for a colour walks the whole document: all kinds of pages and flows.

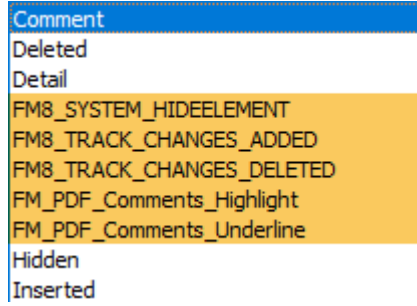
Note: *Be aware that the use of a colour may be found even if you do not recognise it on the particular object. This is because neither the pen (stroke for border) nor the fill pattern may be defined or is 'empty'.*

Conditional Tag →

If a text is flagged with conditions **A** and **B**, then it will be found by searching for **A** or searching for **B**.

To search Conditional Text you must use the FM F/C panel.

The catalogue of conditions tag contains some names which can not be deleted (highlighted in the graphic). These were introduced with FM-xx:



Empty Text Line

Mostly these are a hidden sources of undefined fonts. The script does not remove them.

Searching for empty TextLines walks the whole document: all kinds of pages and flows.

The indication of the Text Line is somewhat difficult here: you see only the text cursor, not the selection handles:

ne, grouped with two others are empty, just 1 blank.



A Indication of found Empty Text Line.

B Indication of found Empty Text Line which is within a group. You may need to refresh the screen (CTRL+L) to see this cursor blinking for better recognition.

Delete before Next This checkbox (**M**) appears only for Find Type Empty Text Line. If set, the currently found object is deleted before the next is searched.

Note: *To delete all empty textlines you may to intitiate a delete cycle (Find, Find Next, ...) multiple times until you immediately get the message “nothing found any more”.⁶*

Overflow Cell

Found overflow is not eliminated. This must be done by the user. With **Find next** you can walk through the whole document.

Searching for overflown cells walks the whole document: cells are search in tables on all kinds of pages and flows.

Overflow Text

Found overflow is not eliminated. This must be done by the user. With **Find next** you can walk through the whole document.

Searching for overflown text frames walks the whole document: all kinds of pages and flows.

⁶ I have not yet found out the reason for this.

Find types not implemented

Character Format...

This search is not handled in the script. The complexity would be too high for me. The setting can just be saved and transferred to the FM F/C panel.

This fact is indicated in the information area.

Conditional Text ...

It turned out that the method to find text with multiple tags does not work in ExtendScript. Hence the script searches for one particular tag only. See [Conditional Tag](#) → on page 7.

Text & Character Formats on Clipboard

This search is not handled in the script. The complexity would be too high for me. The setting can just be saved and transferred to the FM F/C panel.

This fact is indicated in the information area.

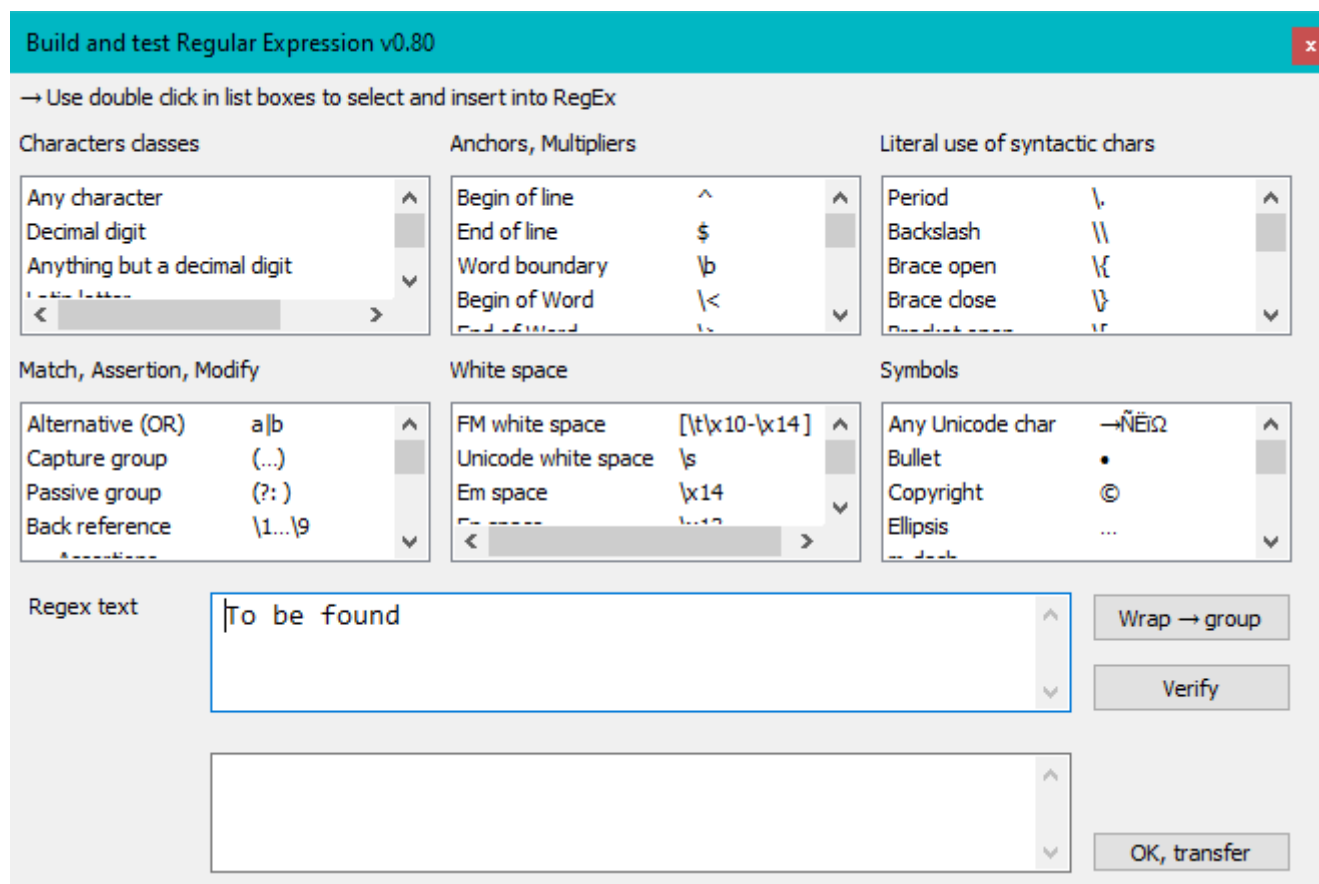
FMsearchReplace.xml

This is just a short example how the file looks like.

```
<saves>
  <item>
    <name/>
  </item>
  <item>
    <name>01 TOC number convert spaces → TAB</name>
    <info>TOC entries starting with arabic or roman numerals are handled</info>
    <findtype>Text:</findtype>
    <findstring>^([0-9]+|[IVXLCDM]+ ?\)?</findstring>
    <options searchMode="2" replMode="0" word="0" case="0" back="0" clone="0"/>
    <replstring>$1\t</replstring>
  </item>
  ...
  <item>
    <name>Z07 Find editor note</name>
    <info>Find char tag "ze-editors-note"</info>
    <findtype>Character Tag →</findtype>
    <findstring>ze-editors-note</findstring>
    <options searchMode="0" word="0" case="0" back="0" clone="0" replMode="0"/>
    <replstring></replstring>
  </item>
</saves>
```

Handling Regular Expressions

FM does not give any feedback about the validity of a RegEx. An incorrect RegEx just triggers the error message “not found” - which of course is misleading. Hence I have implemented a check of the Regular expression.



The field **Regex text** may be populated with text from the field **Find what** of the main panel.

- ▶ Type text directly into the field **Regex text**.
- ▶ To insert a RegEx construct, place the cursor in the RegEx text field and select an item from one of the list boxes. To avoid accidentally selection, you must use double click to insert the item.
- ▶ You convert a selection in **Regex text** into a capturing group by using the **Wrap → group** button.
- ▶ If you think your RegEx is complete, you may check it with the **Verify** button. This checks only validity, not the functionality for the envisioned task. You get feedback from the test in the **Info** area.
- ▶ Use button **OK, transfer** to transfer the created RegEx to the **Find what** field of the main panel. This panel will then close.

A tutorial about RegEx

In march 2016 Marc Pawelec presented a tutorial on RegEx for FrameMaker in blogs.adobe.com/techcom/. This has since then disappeared. But you can [find the blog text here](#).

It seems that he has not tested all the constructs - you need to observe the constraints in the following table, which is the basis of the RegEx panel.

RegEx constructs in FrameMaker

The following constructs have all been tested in FM-15. Take special care to the **highlighted** constructs

Regex	en terminology ^a	Example	Finds	Does not find
Character classes				
.	Any character	. {3}	Hey_whatever\$you will 783	Hey_whatever\$you will 783 (number characters: 8×3+1)
[ab]	Any of the listed	[a\ \$]	Purchase: 17.- \$	Purchase: 17.- \$
[^ab]	Any not in the list	[^a\ \$]	Purchase: 17.- \$	Purchase: 17.- \$
[a-z]	Any in the given range.		Purchase: 17.- \$	Purchase: 17.- \$
\d	Decimal digit		Purchase: 17.- \$	Purchase: 17.- \$
\D	Anything but a decimal digit	\D+	Purchase: 17.- \$	Purchase: 17.- \$
\w	Word character (letter, digit, underscore)	\w+	Combined_Words, figures 123 etc.	Combined_Words, figures 123 etc.
\W	Anything but a word character		Combined_Words, figures 123 etc.	Combined_Words, figures 123 etc.
Anchors				
^	Begin of line			
\$	End of line			
\b	Word boundary	\baesthetic\b	an aesthetic person	anaesthetics
\<	Start of word;	\<a	this is an alphabet	manipulated
\>	End of word	a\>	in a caravanseria	manipulated
\B	Not at a word boundary			
Assertions				
(?=)	Positive look ahead	memo (?=Q)	memoQ	memory
(?!)	Negative look ahead	memo (?!Q)	memory	memoQ
(?<=)	Positive look behind	(?<=k)ot	pot	kot
(?<!)	Negative look behind	(?<!k)ot	kot	pot
Multipliers				
*	0 or more (greedy)			
+	1 or more (greedy)			
?	1 one 0 (greedy)			
*?	0 or more (lazy)			
+?	1 or more (lazy)			
?=	1 one 0 (lazy)			
{n}	n times			
{n,m}	from n to m times			
{n, }	n times and more			
Capturing group				
a b	Alternatives (or)			
(...)	Numbered capturing group			
(?:)	Passive group (not numbered)	^(?:\w+)	First word will be captured, but can not be referenced	
(?<name> ...)	Named capturing group, e.g. (?<abc>[a-z]+)			
\0...\9	Back reference to captured group ^b			

2021-05-29

E:_DDDprojects\FM-FindRep\Docu\FMFindRepl.fm

Regex	en terminology ^a	Example	Finds	Does not find
<code>\k<abc></code>	Back reference to named capturing group, e.g. (<code>?<abc>[a-z]+</code>) <code>\k<abc></code>			
<code>\$0...\$9</code>	Reference in replacement			
	Modifiers			
<code>(?-i:)</code>	case sensitivity ON			
<code>(?i-:)</code> <code>(?i)</code>	case sensitivity OFF			
<code>(?s)</code>	Dot includes line breaks			
	White space			
<code>[\t\x10-\x14]</code>	FM white space chars	<code>[\t\x10-\x14]</code>	TAB Required space, ThinSpace, Numeric Space, m space, n space	Unicode white space character
<code>\s</code>	Unicode white space chars	<code>\s</code>	blank, soft- and hard line break. Unicode white space characters except TAB	FM white space characters
<code>\x14</code>	m space	<code>\x14</code>	m space	
<code>\x13</code>	n space	<code>\x13</code>	n space	
<code>\x10</code>	Numeric space	<code>\x10</code>	Numeric space	
<code>\x11</code>	Non breaking space	<code>\x11</code>	Non breaking space	
<code>\x12</code>	Thin space	<code>\x12</code>	Thinspace	
<code>\t</code>	TAB	<code>\t</code>	Before TAB	
<code>\x09</code>	Line break soft ¶	<code>\x09</code>	Before soft line break after that.	
<code>\x0a</code> <code>\n</code>	Line break hard ¶	<code>\x0a</code> <code>\n</code>	Before hard break After that	
<code>\r</code>	Carriage return	<code>\r</code>	Soft line break from a text import (x0D):expected on next line	this one here next line
	Characters			
<code>ÑËïΩ</code>	Any Unicode character	<code>α</code>	Γαα	Γαα
<code>\u####</code>	Unicode character UCN	<code>\u03B1</code>	Textual u03B1	lower case character alpha: α
<code>\p{L}</code>	Any letter from any language	<code>\p{L}</code>	Nein, es gibt keine 2. oder 3. Erde (Γαα)	Nein, es gibt keine 2. oder 3. Erde (Γαα)
<code>—</code>	m-dash (char, not <code>\xd1</code> , <code>\m</code> or <code>\u2014</code>)			
<code>-</code>	n-dash (char, not <code>\xd0</code> , <code>\=</code> or <code>\u2013</code>)			

a. German and French terminology is reflected in the files `FMfindRepL_de.xml` and `FMfindRepL_fr.xml`.

b. `\0` refers to the whole match

These RegEx do not work in FrameMaker

Marek Pawelec paper claims that these work in FM - but they do not.

Text	en terminology	Does not find
<code>\p{Z}</code>	Any white space or separator	Anything considered white space
<code>\p{Sm}</code> <code>\p{Sc}</code> <code>\p{Sd}</code>	Math symbols, currency symbols, dingbats	Cambria Math: √ ∫ ∏ Segoe UI: € ¥ £ ₹ Segoe UI Symbol: ☞ → ↻ ↺

Text	en terminology	Does not find
\p{N}	Numbers	1234
\p{P}	Punctuation marks	,. : ; { } [] () etc.

Ordinary text search required

There are some special notations which look like RegEx, but are not. They work only in ordinary text search - and in case of the Automatic Hyphen require to set the Find Type accordingly.

Text	en terminology	Example	Finds	Does not find
	Automatic hyphen		This is hyphenated	
\b	bullet (char, not	\b	• \xa5 (•) or \u2022 (•)	
\-	Optional hyphen	\-	hyphenated	not hyphenated
\f	End of Flow	\f	End of cell is the same as end of flow	
\+	non breaking hyphen		non-breaking hyphen	ordinary hyphen

2021-05-29

E:_DDDprojects\FM-FindRep\Docu\FMFindRepl.fm



Transfer to the FM F/C panel

Using the Transfer button creates an intermediate text file which contains symbolic key strokes and human readable comment⁷⁾.

```
; 2021-03-18 16:44 15 en FMfindRepl.cfg generated by FMfindRepl.jsx
; The FM Find/Change panel is controlled by simulated keystrokes.
; Settings name: Experiment!
{Esc}ef                ; Initiate Find dialogue
;*0500                 ; wait 1/2 sec
!d                     ; Look in document
!p                     ; Set Simple Search to enable Clone
!l+{Tab}               ; Go to Replacement text
#$1\t
!b{Tab}                ; Go to Initial line in Replace mode
b                      ; Replace by pasting
!o                     ; Option Consider case
{Esc}-                 ; Clear the check box
!w                     ; Option Whole Word
{Esc}+                 ; Set the check box
!b                     ; Option Find backwards
{Esc}-                 ; Clear the check box
!d                     ; Look in document
!p+{Tab 2}{Home}      ; Go to Initial line in Filetype
{Click A}              ; Open Find Type list for selection ...
p{Down 1}              ; Paragraph Tag →
{Enter}                ; Select current list item
!p+{Tab}               ; Go to Find text
#^([0-9]+|[IVXLCDM]+ ?\)?
!i                     ; Option Wildcard
; --- end of actions
```

You may note here a strange sequence of keystrokes. The reason for this is the dependence of some settings from others.

A subsidiary program - a compiled AutoHotKey script - reads this file and sends keystrokes to the active FM document.

Note: *Do not use your mouse (or tablet pen) during this operation!*

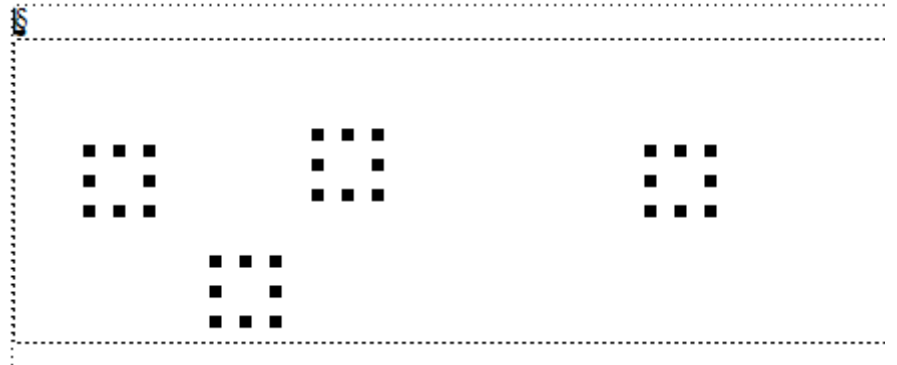
If all settings are transferred, the AHK script issues a completion message.

⁷ This mainly for the purpose of debugging.

Known issues

Empty TextLines

If there are multiple empty text lines within a frame, their indication (visible handles) do not disappear with the next one found:



The 'selection' disappears only after entering another frame. As soon as you activate the document window, the 'selection' disappears...

See note at [Empty Text Line](#) on page 8.

Deletion of empty TextLines

2021-05-29

E:_DDDprojects\FM-FindRep\Docu\FMFindRepl.fm

