

# 52

## Find/Replace extended

### Overview

The [FM community](#) and also the FM [bug base](#) request ability to save and retrieve settings [2020-09-22].

**FMfindRepl** fulfils this demand at least partially.

### Contents

Introduction .....	2
Installation.....	2
Avoid Windows Defender intervention.....	3
Details of the D+DD script installation.....	4
Script invocation .....	5
Menu entries.....	5
Intended work flow .....	5
Working with FMfindRepl .....	6
Special find types.....	11
Special replacements .....	12
Find types not implemented .....	14
Replace modes not implemented .....	14
Search with Wild Card.....	14
FMsearchRepl.xml .....	15
Handling Regular Expressions.....	16
Concerning Undo.....	19
Use Cases.....	20
Known issues.....	21
Older FM versions.....	22
Implementation details .....	23
Other scripts related to find and replace .....	25

#### Important

- Many operations performed by a script can not be undone! Best practice is to save the document before performing script actions on it.
- In case of an error or change of Your mind you can simply Revert to Saved.
- Only scripts which do not modify document contents (e.g. creating a new document) are safe in this respect.

# Introduction

## Use cases

- Find that nasty “inconsistent colour definition” - which does not go away with import from a ‘good’ document. See [Find unwanted colour](#) on page 20.
- Find empty text lines which use an unwanted font. See [Empty Text Line](#) on page 11
- Why is this table cell not filled completely? See [Overflow Cell](#) on page 11.
- Are there any tables or anchored frames using the feature [UserString<sup>1\)</sup>](#)? [User String](#) on page 12.
- Why are recently added index items not in the generated index? See [Change marker type](#) on page 12.
- I want to replace *this text* by a variable which is already defined - but I’m tired of pasting... See [Replace text by a variable](#) on page 13.
- ...

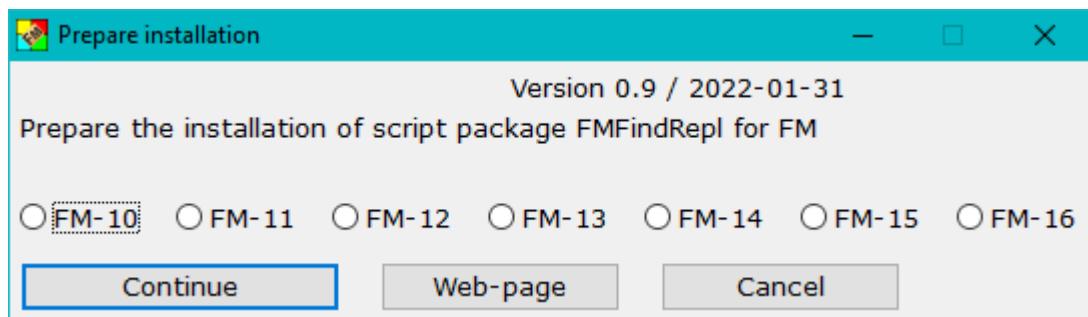
## Installation

- 1 Close FrameMaker if necessary.
- 2 Download the ZIP from the [website](#) and expand it to a location of your liking.
- 3 Execute the file PrepareInstallation.exe (run as Administrator)

**Note:** *Windows Defender may react to this program with a virus warning. This is a false positive. You may check this with an [upload to Virustotal](#). See [Avoid Windows Defender intervention](#) on page 3.*

The language of the installation dialogues adapts to the UI language of your current FM installation (de, en, fr).

- 4 All FM-versions present in your %appdata%/Adobe/FrameMaker/ are listed in the dialogue (on your system there may be only one).

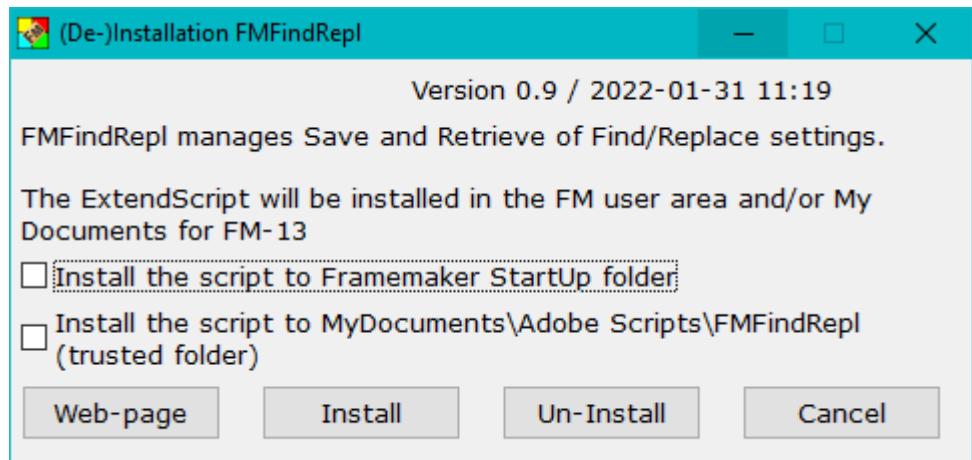


- 5 All FM-versions present in your %appdata%/Adobe/FrameMaker/ are listed in the dialogue (on your system there may be only one).

---

1 Script suite **FMgraph** uses this feature for identification of data tables and diagrams (anchored frames).

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 sub-folder **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.

## Avoid Windows Defender intervention

- 1 In Windows settings navigate to **Windows Security > Virus & Threat protection settings > Manage Settings**
- 2 Go to **Exclusions**
- 3 Add the folder you need to exclude:  
C:\Users\username\AppData\Roaming\DD+DD

## Details of the D+DD script installation

- Unpacking creates a program (`Prepinstalation.exe`) and a directory (`InstallThis`).
- `Prepinstalation.exe` should be executed with admin rights (right-click and select), otherwise a message will appear indicating this.
- `Prepinstalation` adds the information about the current directory (e.g. `C:\Users\Klaus\Desktop\InstallThis`) and the installation directory of the selected FM version to the `InstallThis\config.ini` file.
- Then `InstallThis\Inst-FMfindRepl.exe` is executed, which creates a directory `D+DD\FMfindRepl` in the Windows Start directory and copies everything from `InstallThis` there. In addition, a short cut is created to this directory.
- This directory contains `Inst-FMfindRepl.exe`, which is now executed and the unpacked files that are no longer used (e.g. on the desktop) are deleted.
- `Inst-FMfindRepl.exe` asks where the script should be copied to (Startup folder or `MyDocuments\Adobe Scripts\FMfindRepl`). The selected option must also be selected during a de-installation in order to de-install at the correct location.
- The script consists of the main module `FMmarkers.jsx` and the modules in the `FmfindRepl` directory.
- Various scripts (e.g. the installation of the tool bars) also write something in the `maker.ini` in the user area in the section `[D+DD]`.

# Script invocation

## Menu entries

Document menu

Book menu

### Documentation

The script establishes an entry in the edit menu<sup>2)</sup> with short cuts:

Edit	Shortcut
Find/Replace extended	ESC q f d
Edit	Shortcut
Find/Replace extended	ESC q f b

This pdf is displayed with your default PDF viewer by the **Help** button on the panel.

## Intended work flow

- The user sets up the Find (and Replace) details in the script panel and test the settings in a document or book.
- The current settings can be saved with a name in the script panel.
- Later in time settings can be retrieved, probably modified and executed.
- Available Find and Replace options depend on the type of Search item (Find Type). See *Implementation details* on page 23.
- 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).

Since it becomes very difficult<sup>3)</sup> to handle a translated and thus differently ordered list of modes and options these are represented by check boxes or radio buttons rather than drop down lists.

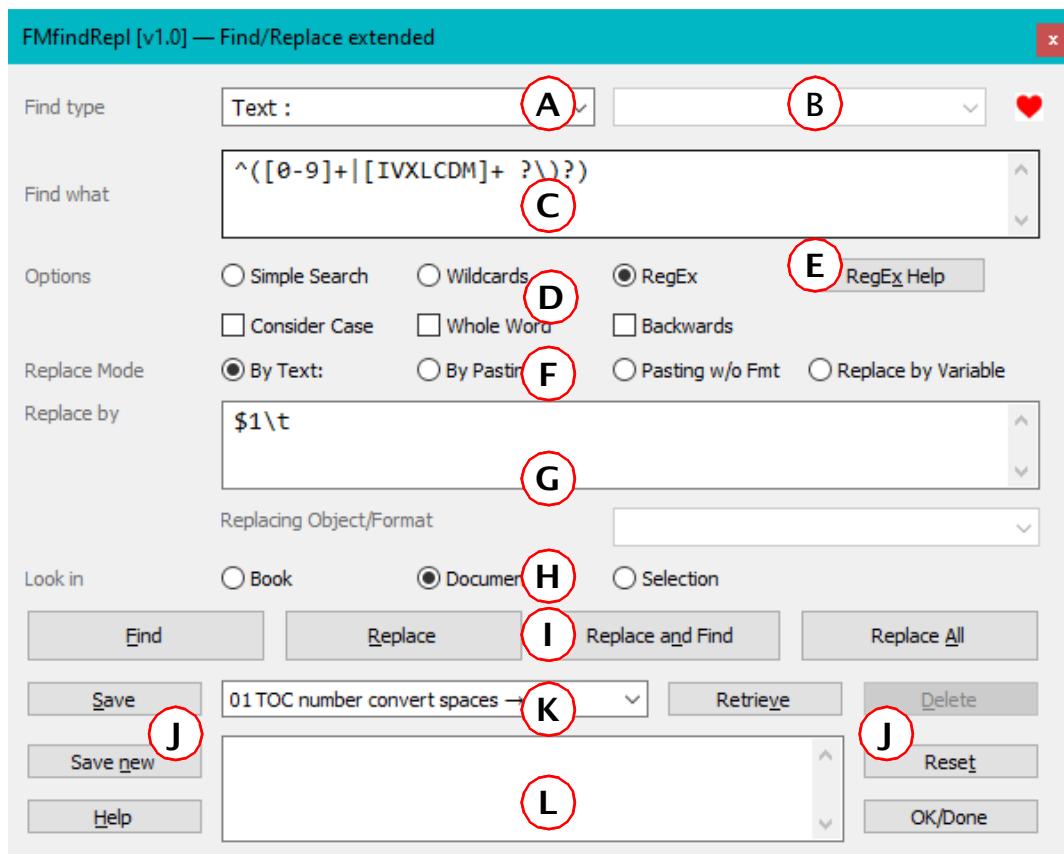
### A note on the User Interface

2 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.

3 For a better programmer of course this would not be a problem...

# Working with FMfindRepl

- 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)

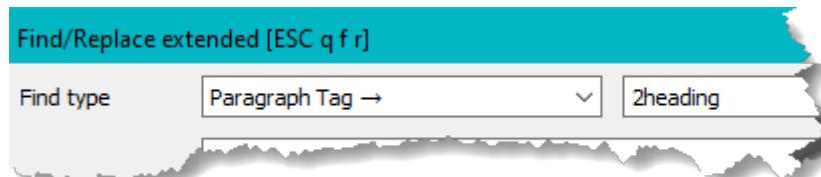


## Find Type (A)

Drop down list of the types Anchored Frame ... Variable named →. Some of the entries have different labels to the original FM F/C panel. Few are left out and some are added (label contains an \*).

## 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 →
  - Colour \* →
  - Condition Tag \* →
  - Cross-Reference of Format →
  - Marker of Type →
  - Object Style Tag →
  - Paragraph Tag →
  - Table Tag →
  - Variable named →
- Both user variables and system variables are listed in alphabetical order.

Special find types

**FMfindRepl** offers some additional find types compared to the FM Find/Change dialogue. See [Special find types](#) on page 11.

Find types not implemented

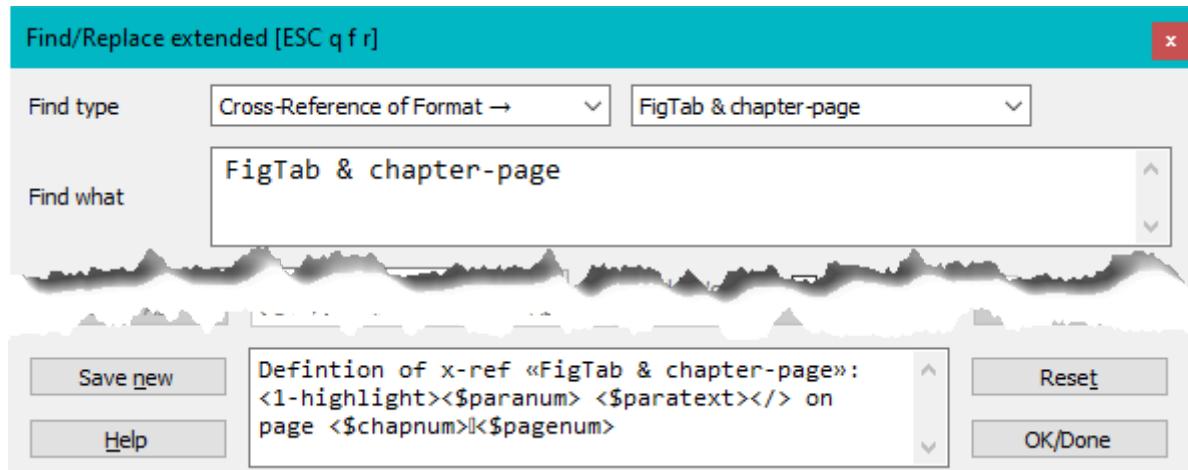
Due to lack of knowledge or information, some find types present in the standard FM Find/Change panel are not implemented. See [Find types not implemented](#) on page 14.

Additional information

For the following Find Types information about the found object is provided in the info area L:

- Cross Reference Format (see screen shot)
- Marker of Type: Marker text
- Variable: Definition

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)

**Note:** Enabling of Find Option depends on the selected Find Type. See [Main control table](#) on page 23.

### Radio buttons

#### Simple Search

This is the most used case and available for all supported Find Types.

#### Wildcards

See [Search with Wild Card](#) on page 14.

#### RegEx

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

The created RegEx will be inserted into the edit field **Find what**. Only elements for the Perl flavour of RegEx are implemented. Hence you get a warning if your installation defines another flavour.

#### Note:

If you do not use the button **RegEx Help** to create the regular expression, this is not tested.

### Check boxes

#### Whole Word

If the search string shall match only full words, check this. *Disabled for RegEx*.

#### 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. *Disabled for RegEx*. See [Backward search may halt FM](#) on page 21.

**Button RegEx Help (E)**

This becomes active only for Option Regex. It opens a panel from which Regular Expression elements can be selected.  
See [Handling Regular Expressions](#) on page 16.

**Replace mode (F)**

**Note:** Enabling of Replace Modes depends on the selected Find Type. See [Main control table](#) on page 23<sup>4)</sup>.

**Radio buttons**

Replacing **To Character Format...** is not implemented. Use the Standard FM dialogue for this.

**By Text**

This is a default for most Find Types.

If **RegEx** is active, the **Replace by** entry may refer to capture groups with the symbolic notation  $\$n$ . See [Reference to capture groups](#) on page 19.

Editors note 1

It has to be checked which format this implies.

**By Pasting**

The selection is replaced by the contents of the fully formatted clip board.

**Pasting w/o Fmt**

The clipboard is cleared from local formatting. Markers and variables are preserved. The clipboard must not contain more than one paragraph. Hence do not include a paragraph end mark in the selection at copy time.

**Special treatment**

To replace an object or property with catalogued name the possible replacements are provided in the drop-down list [Replacing Object/Property](#).

This caption is changed for certain Find Types to indicate the real function<sup>5)</sup>:

**Change Marker Type** for Marker (any).

**Delete Object** for Empty Textline.

**Replace Colour** for Colour.

**Remove Override** for Character Format, Object Style, Paragraph Format and Table Format.  
See also [Remove Object overrides](#) on page 21.

**Replace Obj Text** for Marker of Text, Marker of Type and User String. The Marker text or User String is replaced by the given string.

**Replace Obj Format** for Any Cross-Reference, Cross-Reference of Format, Any Table, Table Tag. Since The format names are catalogued, you select the replacement from the drop-down list Replacement Object.

**Replace by Variable** for Text. You can select the variable name from the activated drop-down list Replacement Object.

See [Special replacements](#) on page 12 for further details.

**Replace by (G)**

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

4) Clone Case is not implemented, because the detailed function of this is not clear to me.

5) Standard FM Find/Change replaces the objects by simple text, which is IMHO not a reasonable action.

## Replacing Object

To remove found text, the replacement is empty (replace by nothing).

If the Find Type is an object with catalogued names (e.g. Cross Reference), then the adjacent drop-down list will be filled with same items as the drop-down list right to Find Type.  
If you want to delete the contents of the object, you select the top empty item from the list.

## Look in (H)

### Radio buttons

#### Document

This is the default and the only possible selection for **FMfindRepl** version 1.0

#### Book

If panel is opened while book is active, then this is checked.

#### Selection

Editors note 2

Search in selection must be evaluated. The FM F/C process finds only once - and the selection is no more active.

## Action Buttons (I)

#### Find

Find the text or object according to the settings ([Options \(D\)](#) on page 7).

#### Replace

Replace the text or object according to the settings ([Replace mode \(F\)](#) on page 8).

#### Replace and Find

<sup>6)</sup>Same as pressing Replace and then Find.

#### Replace All

<sup>6)</sup> Use this with care (you get a warning). The Find / Replace function is performed until Find is no more successful.

## Maintain the settings (J)

#### Note:

*Although you can transport (e.g. copy) the file FMfindRepl.xml between FM installations - it is not possible to handle such a file created with a different UI language of **FMfindRepl**.*

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.

6 Not implemented in version 1.0

7 The data is kept in file FMfindRepl.xml.

- 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 \$, @ [ etc) 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

(Formatted for better reading)

```
01 TOC number convert spaces → TAB
Info          TOC entries starting with arabic or
               roman numerals are handled
Find type     Text :
Find what      ^([0-9]+|[IVXLCDM]+ ?\))?
Options        RegEx
Replace by    $1\t
```

### Delete item in list

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 it: copy %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 :

- Info from the retrieved settings.
- Properties for some Find Types (e.g. the text of a found marker).
- 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

2022-06-29

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

[D+ D-]

## Rationale for these find types

In the drop-down list of Find Types these are listed with an \*.

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 or misnamed colour.

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, flows and objects. Hence this may take some time!

#### 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 with) nor the fill pattern may be defined or is 'empty'.*

### Condition Tag →

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

The catalogue of conditions tag contains some names which are used by FM and can not be deleted (FM8\_... and FM\_PDF\_...). These were introduced with FM-15. In the catalogue of **FMfindRepl** these are filtered out.

#### Note:

*Search for Conditional Text is not implemented, you must use the standard Find/Change dialogue.*

### Empty Text Line

Mostly these are a hidden sources of undefined fonts.

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

- ▶ Select **Text Line**: as Find type.
- ▶ Leave the **Find what** field empty.
- ▶ Use the **Find** button.
- ▶ You can remove this unwanted object by button **Replace**.

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.

|  
Found within a group



Found as single Text Line

### Overflow Cell

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

No Replace option is provided. It is left to the user what to do with the found object.

### Overflow Text

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

No Replace option is provided. It is left to the user what to do with the found object.

## User String

There is a long list of objects which can have property `UserString`. **FMfindRepl** searches the strings only in the following objects: Doc, AFrame, Tbl.<sup>8)</sup> **FMgraph** uses the `UserString` property for diagrams (anchored frames) and data tables.

# Special replacements

## Change marker type

This function was developed to cope with the frequent error of missing index entries. It often happens that one intends to create an index entry, but has not selected the proper marker type in the marker dialogue.

- ▶ Use Find **Marker (Any)**. If you have found a text which obviously should be an index entry (for example: not a cross reference), then
- ▶ Activate the radio button **Change Marker Type**
- ▶ In the down list Replacement Object select the new marker type to be used. Its name will appear in the **Replace By** field also.
- ▶ Use the **Replace** button.

## Change object format

The format of objects which is catalogued (Character, Condition Tag, Cross Reference, Paragraph, Object, Table, Variable) can easily be changed:

- ▶ Activate the radio button **Replace Obj Format**
- ▶ In the down list Replacement Object select the new format to be used. Its name will appear in the **Replace By** field also.
- ▶ Use the **Replace** button.

**Note:** *If you change the format of an unresolved cross-reference, then this reference will be removed.*

## Define user strings

When searching for the first time in a document - most, if not all - of these strings may be empty. To define `UserStrings`:

- ▶ Use Find Type User String
- ▶ To find empty user strings (yet undefined) type ??? into the **Find What** field.
- ▶ Activate the radio button **Replace Obj Text**
- ▶ Fill the edit field **Replace By** with the desired text.
- ▶ Use **Replace** button.

## Remove colour

To remove the colour from a found object

- ▶ Activate the radio button **Replace Colour**
- ▶ Use **Replace** button.

## Remove property

It is possible to remove the following properties from an object which has its properties catalogued:

- Character Tag ???
- Condition Tag ???
- Object Tag ???

To remove such a property

---

<sup>8)</sup> On user demand this list could be extended (global variable `aUserStrTypes`, function `CollectUserStrings`).

- ▶ Select the top empty item in the drop down list **Replacing Object/Format**
- ▶ Use **Replace** button.

## Remove textural property

Editors note 6

You can clear the text property of the following objects:

- Cross Reference

**Marker of Text ...**

- User String

To clear such a property,

- ▶ Activate the radio button **Replace Object Text**
- ▶ Clear the field **Replace By** i
- ▶ Use **Replace** button.

## Replace text by a variable

After you found the text, you may replace it by a variable:

- ▶ Activate the radio button **Replace by Variable**
- ▶ In the down list Replacement Object select the variable to be used. Its name will appear in the **Replace By** field also.
- ▶ Use the **Replace** button.

## Replace text in a Marker

Editors note 7

After you found the marker (by type, by text) you want to change:

- ▶ Activate the radio button **Replace Obj Text**.
- ▶ In field **Replace By** type the replacement text.
- ▶ Use the **Replace** button.

You can change the text property of the following objects:

- Cross Reference

**Marker of Text ...**

- User String

- ▶ Activate the radio button **Replace Object Text**
- ▶ Clear the field **Replace By**
- ▶ Use **Replace** button.

## Find types not implemented

Selecting one of these Find Types informs you with a message that **FMfindRepl** does not handle this type. Use the standard FM dialogue.

### Character Format...

The complexity is to high for me.

### 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 [Condition Tag →](#) on page 11.

### Publisher

This is a DITA element and should not be listed in unstructured FrameMaker... Hence it is left out in the drop-down list.

### Text & Character Formats on Clipboard

The complexity is to high for me.

## Replace modes not implemented

I have no experience with Structured FrameMaker. Hence I have not implemented the following Replace Modes:

- Attribute Name To:
- Attribute Value To:
- Element Tag To:

I do not understand what the following really means and have it not implemented:

- To Character Format ... In FM you can select an Anchored frame and replace it by a Character Format - IMHO quite nonsense.

## Search with Wild Card

FM Help misses the important information **highlighted here**.

Wildcard	Description	Sample
*	Star. Any number of characters (blank is not considered a character!)	*orm searches for: form or inform
	Pipe. Any number of Spaces or punctuation	This includes the ¶ end character!
.	Dot. Any one character	fo .m searches for form or foam
^	Caret. Start of a line	
\$	End of line	This includes the ¶ end character!
[ab]	Any one of the bracketed characters	f[ao]rm searches for farm or form
[^ab]	Any character except the bracketed characters	f[^ao]rm searches for f followed by any character except the a and o followed by rm
[a-f]	Any character in the specified range	f[b-e]rm searches for farm or form but not, for example, ferm

The is no equivalent to the capture groups of RegEx for Wild Card finds. Hence I mostly use the RegEx route.

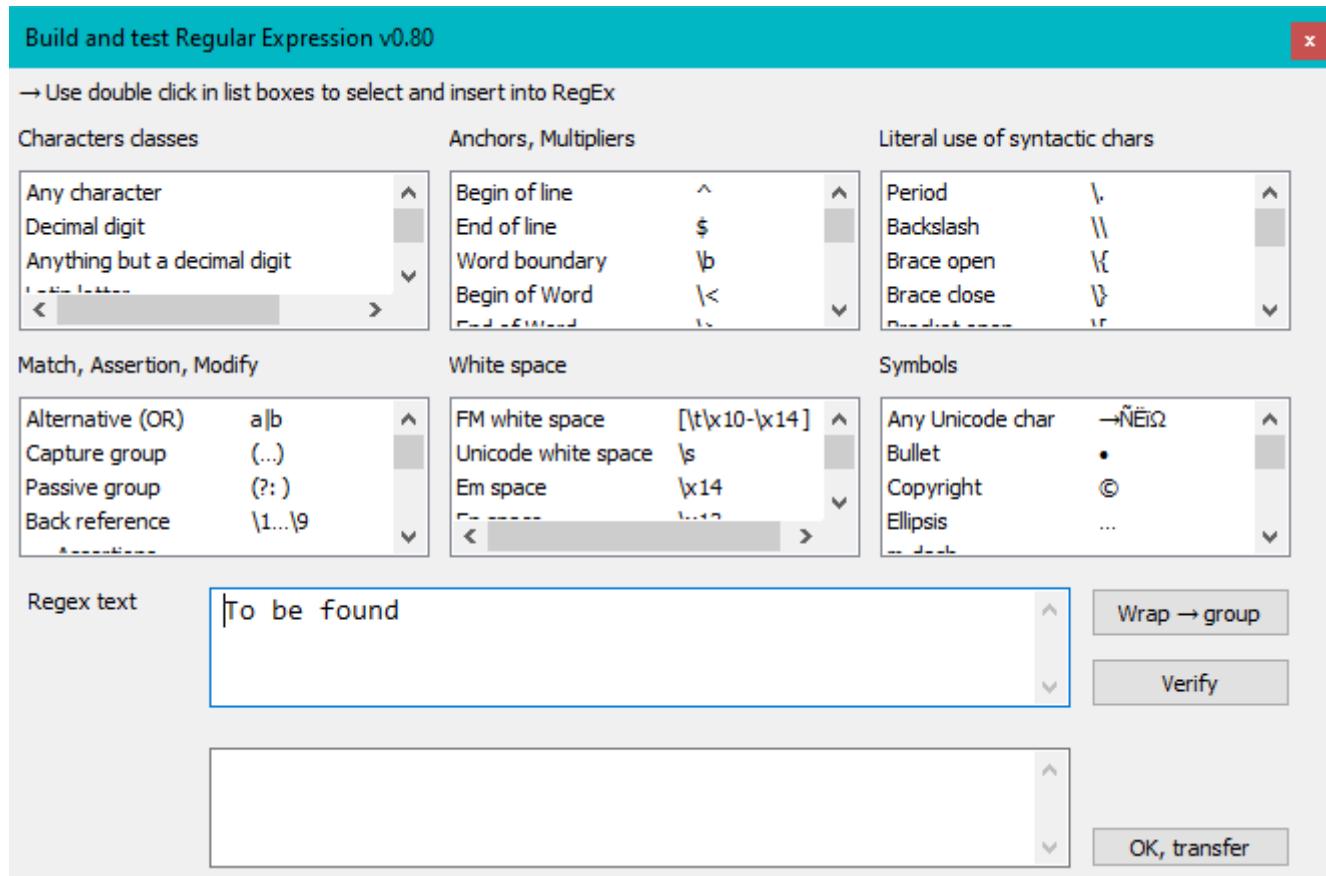
## FMsearchRepl.xml

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" word="0" case="0" back="0" replMode="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" 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 (rudimentary) 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/](http://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 <sup>a</sup>	Example	Finds	Does not find
<b>Character classes</b>				
.	Any character	.{3}	Hey_whatever\$you will 783	Hey_whatever\$you will 783 (number characters: 8x3+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.
<b>Anchors</b>				
^	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			
<b>Assertions</b>				
(?= )	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
<b>Multipliers</b>				
*	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			
<b>Capturing group</b>				
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 <sup>b</sup>			

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

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's paper<sup>9)</sup> claims that these work in FM - but they do not.

## 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	\-	hyphen-ated	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

## Reference to capture groups

For the replacement process the back references are noted with the \$-sign:

\$0 Refers to the whole match

Refers to the first capture group

**\$9** Refers to the 9<sup>th</sup> capture group.

## Editors note 8

Selected text  
Find what  
Replaced by

This shall be illustrated with an example.

# Concerning Undo

It turns out that in many cases (even multiple) undo after find/replace is possible. The detailed circumstances for this welcome behaviour have not been researched.

Hence do not rely on it.

9 <https://blog.adobe.com/en/publish/2016/03/10/framemaker-regular-expressions>

# Use Cases

## Find unwanted colour

When updating a book you may get the message for “inconsistent colour definitions”. You have done the usual import from a ‘good’ document to fix this - but the message does not disappear.

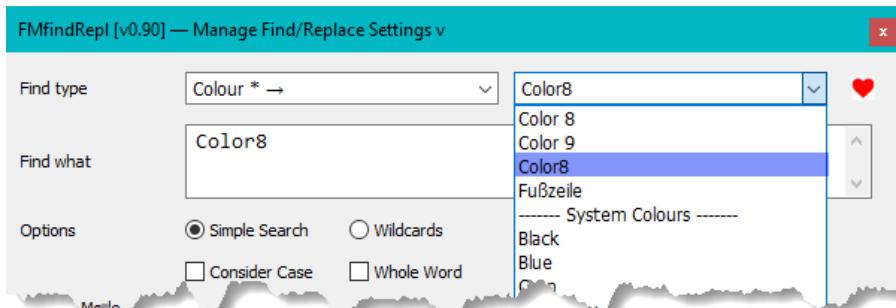
It may be that simply an unknown colour is defined. For example, in a book file Color8 may have been defined and used, but it should be Colour 8 (watch the blank).

This problem can not be solved with standard FM methods:

- Looking through the colour catalogue in each file you may find the one with the unwanted colour.
- Searching for a custom character format (all `asis`, just colour set) does not find a ¶ with no added character format.
- Searching for a ¶ format with all `asis` but colour specified is not possible at all.

To find the culprit, do this:

- 1 For each file in the book check the available colours until you find the unwanted colour:



- 2 In this document use **Find / Find Next** to find the object which uses this unwanted colour:

### 17.1.4.1.16 Kontonummer 1¶

Schreibweise der Variablen§	<b>JANWKTO1§</b>
Einsetzbar§	Im allgemeinen Rechnungsbereich und im allgemeinen Bereich der Lieferantenbestellungen§

Variable als Platzhalter für die Kontonummer innerhalb der ersten möglichen Bankverbindung in den Anwenderdaten¶

- 3 The found object is a paragraph, hence either you need to apply a correct ¶ format or change the colour definition in the ¶ format.
- 4 This method can also be used to find an unwanted colour in a diagram - because the colour search walks all objects...

## Known issues

### Backward search may halt FM

Backwards search for the following Find Types may cause FrameMaker to hang. This might be after you continue search when already on the first item in the document.

But there are also other conditions for this fault. This might also happen in the standard FM Find/Change process.

- Cross Reference (unresolved)
- Marker (Any), Marker of Text, Marker of Type
- Rubi (Any)
- Table (Any), Table Format Override; Table Tag
- Variable (Any); Variable named

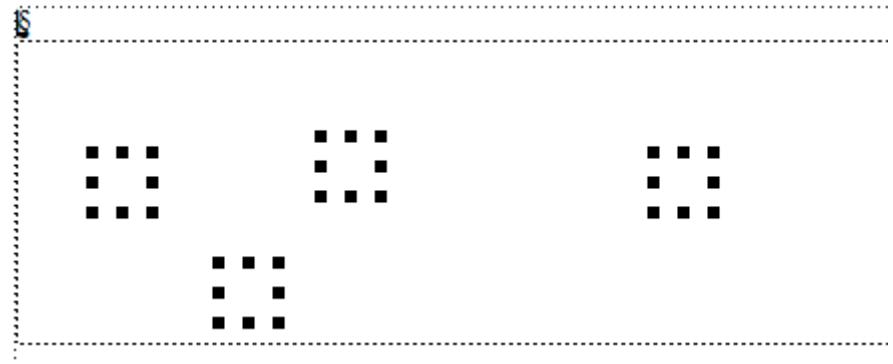
Hence the warning (first paragraph above) is issued for the relevant Find Types.

### Create an empty style

- Have the cursor outside of any object (outside the text frame)
- Use the **Object Designer** and set all properties to *nothing/asis* (As Is, empty fields, cleared check boxes).
- Save for example as `!_None` or `00_Nonthing` to list it at the top of the catalogue.

### 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...

### Remove Object overrides

Re-applying the original style to the following objects removes the override:

- Polygon, circle or other geometric object.
- Copied or referenced image.

Although you can remove the override for these object types, the next Find will still recognise an override. Even starting the Find somewhere after the object will find it again:

- Anchored frame.
- Math object.

**Note:** You may get rid of this nasty situation (for the math objects only) by applying a non-style to the object and then re-apply the wanted style. See [Create an empty style](#) on page 21.

Removal conditions	<p>Successful removals of overrides for object styles depends at least<sup>10)</sup> on these factors:</p> <ul style="list-style-type: none"><li>• The style itself must not contain properties irrelevant for the particular object; for example a run-around setting for a math object (equation).</li><li>• The type of override. The override may be “irrelevant” for the type of object.</li></ul>
--------------------	---

## Older FM versions

Some searchable objects became available in later FM versions. Hence there are appropriate checks and the user is informed about this.

- Object Properties became available with FrameMaker version 11.
- User strings became available with FrameMaker version 14

---

<sup>10)</sup> I say so because many of my attempts to remove the overrides are not successful. The search for an override still finds it, although I have removed...

# Implementation details

For the various find types (what to find) a reasonable set of Find Modes, Replace Modes or even Find Options varies significantly. Hence the process is controlled by a able defining these variants.

## Find Modes

Radio buttons are set in `SetFindMode`.

- 0 Index for the default mode (1...3)
- 1 Simple find method
- 2 Wild card method
- 3 Find by Regular Expression

## Find Options

Check boxes are set in `SetFindOptions`:

- 0 Case relevant search
- 1 Consider whole word
- 2 Find backwards

## Replace Modes

Radio buttons are set in `SetReplMode`.

- 0 Default index
- 1 To text
- 2 By pasting
- 3 Paste without format
- 4 Object dependent treatment.

## Constraints for find and replace

Not all combinations of modes and options are relevant or even dangerous for certain Find Types. Of course, some decisions are my personal view, but most are based on experience:

- I have no clue how to implement **Wildcard** search for the additional Find Types (e.g. colour names).
- Since I do not know how to handle the **Clone Case** in replacement, I have left out this feature.
- For example, in my opinion it does not make sense to replace an anchored frame by text - which is the default replace action in FM. It may however be reasonable to replace it by another frame using the Paste method.

## User strings

These are searched only in the highlighted object types (which can contain user strings): **AFrame**, app, XRef, Var, TblFmt, **Tbl**, SubCol, RulingFmt, Row, PfgFmt, PgF, Marker, Fn, **Flow**, Element, **Doc**, CondFmt, CombinedFontDefn, Color, CharFmt, Cell, BookComponent.

On user demand the list of handled objects could be extended.

## Main control table

Table `asFindType` exists both in the XML files as well as in the module `FMfindRepl_Globals.jsx`. It defines the status enabled of the check box or radio button as well as default settings.

The constants are found in FDK `fapi.h` under `F_ApiFind_` Items. There are no `FS_Find...` for graphic objects. Hence finding for example, (empty) `TextLine` needs another approach.

In the tables T stands for True, F stands for False. The same notation is used in the program tables (XML files).

Type (ordered) <sup>a</sup>	sFType	con- stants	Fmodes	Fop- tions <sup>b</sup>	Rmodes	Comment										
						(label for Special Treatment)										
Items with * are D+DD addenda Items with → open the corresponding catalogue Default is highlighted	Short-hand for Type and constants															
			FS_XXX	FV_XXX	Default index	Simple	Wildcard <sup>c</sup>	Regex	Case	Word	Back	Default index	To text	By Paste	Paste w/o Format	Special treatment
Anchored Frame	ANCFRM	8, 7	1, T, F, F	F, F, T	2, F, T, F, F											
Automatic Hyphen	AUTHYP	8, 10	1, T, F, F	F, F, T	1, T, T, T, F											
Character Format Override	CHRFMO	8, 13	1, T, F, F	F, F, T	4, F, F, F, T										(Remove Override)	
Character Format...	CHRFMT														<b>FM original only</b>	
Character Tag →	CHRTAG	5	1, T, T, T	T, T, T	1, T, T, T, T										(Replace Obj Format)	
Colour * →	COLOUR		1, T, F, F	F, F, F	4, F, F, F, T										(Replace Colour)	
Conditional Tag * →	CONTAG	13	1, T, F, F	F, F, F	4, F, F, F, T										(Replace Obj Format)	
Conditional Text...	CONTXT														<b>FM original only</b>	
Cross-Reference (any)	XRFANY	8, 1	1, T, F, F	F, F, T	1, T, T, T, T										(Replace Obj Format)	
Cross-Reference of Format →	XRFFMT	11	1, T, T, T	T, T, T	1, T, T, T, T										(Replace Obj Format)	
Cross-Reference (unresolved)	XRFUNR	8, 2	1, T, F, F	F, F, T	0, F, F, F, F										Find only	
Empty TextLine : *	EMTTXL		1, T, F, F	F, F, F	4, F, F, F, T										(Delete Object)	
Footnote	FOONOT	8, 8	1, T, F, F	F, F, T	0, F, F, F, F										Find only <sup>d</sup>	
Marker (any)	MRKANY	8, 0	1, T, F, F	F, F, T	4, F, F, F, T										(Change Marker Type)	
Marker of Text:	MRKTXT	10	1, T, T, T	T, T, T	4, F, F, F, T										(Replace Obj Text)	
Marker of Type →	MRKTyp	9	1, T, T, T	T, T, T	4, F, F, F, T										(Replace Obj Text)	
Object Style Format Override	OSTFMO	8, 15	1, T, F, F	F, F, T	4, F, F, F, T										(Remove Override)	
Object Style Tag →	OSTTAG	18	1, T, F, F	T, T, T	1, T, T, T, T										(Replace Obj Format)	
Overflow cell *	OVLCCEL		1, T, F, F	F, F, F	0, F, F, F, F										Find only	
Overflow text *	OVLTXT		1, T, F, F	F, F, F	0, F, F, F, F										Find only	
Paragraph Format Override	PGFFMO	8, 12	1, T, F, F	F, F, T	4, F, F, F, T										(Remove Override)	
Paragraph Tag →	PGFTAG	4	1, T, T, T	T, T, T	1, T, T, T, T										(Replace Obj Format)	
Rubi	RUBANY	8, 11	1, T, F, F	F, F, T	1, T, T, T, F											
Table (any)	TBLANY	8, 9	1, T, F, F	F, F, t	1, T, T, T, T										(Replace Obj Format)	
Table Format Override	TBLFMO	8, 14	1, T, F, F	F, F, T	4, F, F, F, T										(Remove Override)	
Table Tag →	TBLTAG	6	1, T, T, T	T, T, T	1, T, T, T, T										(Replace Obj Format)	
<b>Text :</b>	TXTXTT	1	1, T, T, T	T, T, T	1, T, T, T, T										(Replace by Variable)	
Text Inset (any)	TINANY	8, 3	1, T, F, F	F, F, T	0, F, F, F, F										Find only	
Text Inset (unresolved)	TINUNR	8, 4	1, T, F, F	F, F, T	0, F, F, F, F										Find only	
Text on Clipboard (formatted)	TXTCLP														<b>FM original only</b>	
User String	USRSTR		1, T, F, T	T, T, F	4, F, F, F, T										(Replace Obj Text)	
Variable (any)	VARANY	8, 6	1, T, F, F	F, F, T	1, T, T, T, T										(Replace Obj Format)	
Variable named →	VARNAME	12	1, T, T, T	T, T, T	1, T, T, T, T										(Replace Obj Format)	

a. Publisher (FS\_FindObject (8), FV\_FindAnyPub (5)) has been left out. This is a DITA element and should not be listed in unstructured FrameMaker...

b. Foptions: default = none are set (all are false). Clone case (for replacement) is not implemented.

c. Wild-card characters work only for Find. No equivalent method to the RegEx catch groups exist for Replace.

d. FM F/C changes the footnote reference to text/paste buffer etc. - which is IMHO senseless (the note itself is not removed).

# Other scripts related to find and replace

Since finding and replacing is one of the fundamental tasks in documentation, over time a number of specialised scripts have been developed. It is natural that some functions overlap...

## Find all text concurrencies

[[Rick Quattro](#)] [FindAll](#) allows you to find all occurrences of a string or regular expression match in a FrameMaker book or document. It generates a new document with the found text. Paragraphs contain cross references to the original location. Thus you can go there and judge the validity of the text.

## Find/Change batch

[[Rick Quattro](#)] [FindChangeBatch](#) allows you to perform batches of finds and changes on a FrameMaker document or book. You use a seven-column table in a FrameMaker document to set up your find/change pairs and their settings.

## Find/Change Formats batch

[[Rick Quattro](#)] [FindChangeFormatsBatch](#) is a delightfully simple script that allows you to Find/Change hundreds (or thousands) of FrameMaker formats with a single command. Finds/changes can be performed on the following formats: paragraph, character, table, condition, cross-reference, user variable, master pages, and marker types. [FindChangeFormatsBatch](#) uses a simple FrameMaker table to map your old formats to new ones. The built-in Write Formats Table command gives you a head start on building your table.

This script seems to be the more powerful successor of [FindChangeSpecial](#) (for which all [web references](#) are invalid).

## Path changer

[[Rick Quattro](#)] FrameMaker documents have graphics imported by reference, text insets, and external cross-references, all pointing to files outside of the FrameMaker document. FrameMaker books point to book components that can be located just about anywhere. When these referenced files get moved or renamed, you can end up with a combination of missing graphics, unresolved text insets, unresolved cross-references, or books with missing components.

[PathChanger](#) is a series of scripts that makes it easy to change these paths for a FrameMaker document or book. It has a command for writing these paths to a simple .csv file. This file can be opened with Excel (or a text editor) where you can edit the paths. Another command applies the paths back to the FrameMaker document or book. Additional commands ...

## Text formatter

[[Rick Quattro](#)] [TextFormatter](#) allows you to apply Character Formats to text in batches! You fill in a FrameMaker table with the words or phrases you want to search for and the Character Format that you want to apply to each one. The script flies through your document or book and applies the Character Format to each occurrence. There are options for case-sensitive and whole word searches and you can specify paragraph formats that you want to skip. This script is a huge time saver!

## [i]-frame scripts

itl ([www.itl.eu](http://www.itl.eu)) developed a highly esteemed suite of scripts: [i]-frame. Some of them were even free. Unfortunately they are no more available since 2020.

