



AutoFM User Guide

v.1.0.05
2015-10-05

Leximation, Inc.

AutoFM User Guide.

by Scott Prentice, Leximation, Inc.

Copyright © 2011-2015 Leximation, Inc. All rights reserved.

Published by Leximation, Inc., 122 H Street, San Rafael, CA 94901.

The content of this document is furnished for informational use only and is subject to change without notice. While every precaution has been taken in the preparation of this document, the publisher and author assume no responsibility for errors or omissions, or for damages resulting from the information contained herein.

This document was authored and published using FrameMaker and DITA-FMx.

The most current version of this guide is available on the Internet at
<http://docs.leximation.com/autofm/1.0/>

Adobe, the Adobe logo, Frame, and FrameMaker are trademarks of Adobe Systems Incorporated of San Jose, California, USA.

Contents

Chapter 1:	AutoFM	1
Chapter 2:	Installation and Setup	3
	Uninstalling AutoFM	5
Chapter 3:	The AutoFM Data Model	7
	Sample Script	7
	Troubleshooting	9
	AutoFM Elements	10
Chapter 4:	Create and Run an AutoFM Script	19
Appendix A:	Revision History	21
	1.0.05 - 5 October 2015	21
	1.0.04 - 20 April 2014	22
	1.0.03- 19 August 2012	23
	1.0.02- 27 July 2012	23
	1.0.01- 7 July 2012	24
	1.0.00 - 4 February 2012	25
	Index	27

1

AutoFM

AutoFM provides a scriptable interface to FrameMaker, allowing the automation of assembly, pagination, production, and printing of documents in FrameMaker.

Documentation last updated: October 5, 2015

AutoFM is a client FDK plugin developed for FrameMaker 7.0 and later.

The AutoFM plugin responds to the instructions in an AutoFM XML script file. You can create an AutoFM script using any XML editor. A structure application is provided to make it easy to create and test AutoFM files in FrameMaker.

RELATED INFORMATION:

"Installation and Setup" on page 3

"The AutoFM Data Model" on page 7

"Create and Run an AutoFM Script" on page 19

"Revision History" on page 21

2

Installation and Setup

Install the AutoFM plugin and the AutoFM structure application.

AutoFM relies on a structure application for parsing and analysis of the AutoFM script. After installing the plugin, be sure to install the structure application.

- 1) Run the AutoFM installer application.

This copies the *autofm_<fmver>.dll*, *autofm-pdf.dll*, *autofm.chm*, and *autofm_app.zip* files to the *FrameMaker\Pubs-Tools* folder. It also updates the *maker.ini* file to register the plugin DLLs.

- 2) Install the AutoFM structure application.

If you're using AutoFM with FMx-Auto, and DITA-FMx is installed, you'll be prompted to auto-install the structure application after restarting FrameMaker. If you choose not to auto-install or are using this separate from FMx-Auto, you'll need to manually install the structure application as follows.

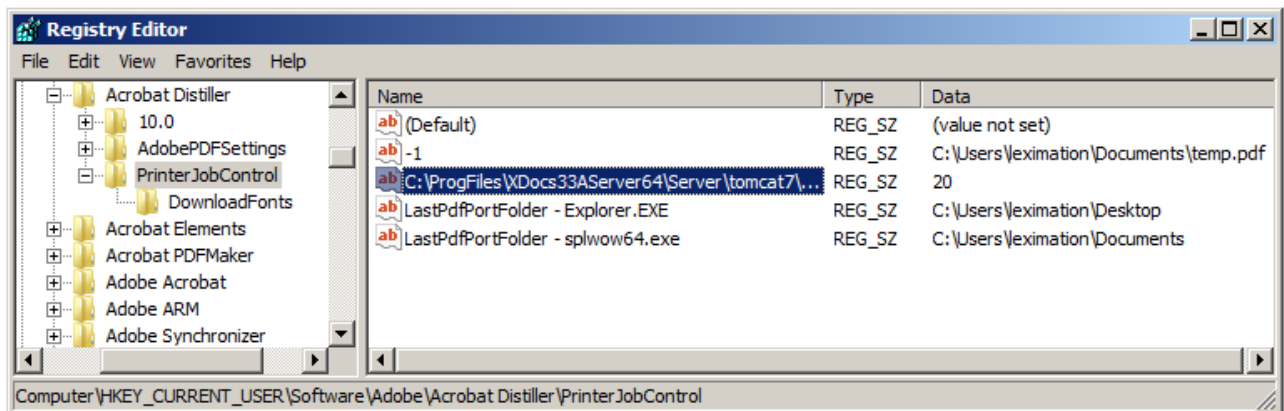
NOTE: *The AutoFM structure applications are provided in FM8 format. If you're installing on FM7.x, you'll need to save to the FM7 format before use. If you are unable to do this, contact Leximation.*

- a) Extract the *autofm_app.zip* archive to the *FrameMaker\Structure\XML* folder so it creates an *AutoFM* folder at that location.
- b) Open your structure application definitions file (**Structure Tools > Edit Application Definitions**).
- c) In the Structure View window place your insertion point just before an existing XMLApplication node, then choose **File > Import > File**.
- d) In the Import dialog, navigate to the *Structure\XML\AutoFM* folder and select the *autofm_structapps-stub.fm* file.
- e) Accept the defaults in the Import Text Flow dialog, and choose the Import button.
- f) Save the file and choose **Structure Tools > Read Application Definitions**, then close the structure application definition file.

3) Review your Adobe PDF Printer Preferences.

If you plan to use the option to “print” to a PDF file, you must make sure the Adobe PDF printer preferences are set properly. The Adobe PDF Output Folder option must be set to “Prompt for Adobe PDF Filename”. If you are running AutoFM on a server or in an automation workflow, you probably don’t want the generated PDF to display on screen. To prevent the PDF from displaying, deselect the “View Adobe PDF Results” option.

Each time you “print” a PDF, an entry is added to the PrinterJobControl key (folder) in the registry (shown in the following screen shot). This may be useful for certain situations, but it can also quickly add to the “cruft” in the registry if you’re generating a lot of PDFs through AutoFM. If you’d like these entries deleted as they are created by AutoFM’s automated processing, set the General/DeletePdfRegEntry parameter to 1 in the *autofm.ini* file. The *autofm.ini* file is located in the *FrameMaker\Pubs-Tools* folder for Windows XP systems and in the *%appdata%\Adobe\FrameMaker\<version>\Pubs-Tools* folder on Windows Vista, 7, and 8 systems.



IMPORTANT: If you are running a 64-bit version of Windows, the PDF generation process relies on the *splwow64.exe* application. By default this is found at *C:\Windows\splwow64.exe*, but if it is in another location on your system, enter the full path and filename in the General/SPLWOW64 parameter in the *autofm.ini* file.

4) Disable automated update checking.

Choose **Pubs-Tools > Check for Updates**, then enter “0” in the Check For Updates Frequency field and choose Close. This prevents the Leximation plugins from running the Check for Updates process which may interrupt automated processing. If you have other plugins installed that perform automated update checking that could interrupt the AutoFM processing, disable those actions as well.

NOTE: Remember to review the *Troubleshooting* section.

It is best to restart FrameMaker after completing the installation of the structure application. You are now ready to create an AutoFM file.

Uninstalling AutoFM

There is no Uninstall application provided for AutoFM. Because only a few files are installed, it's easiest to just perform this uninstallation manually.

To disable the AutoFM plugin without removing any files:

- Comment out the “Pubs-Tools:AutoFM” and “Pubs-Tools:AutoFM-PDF” entries in the APIClients section of your *maker.ini* file in the main *FrameMaker* folder (just add a semicolon at the start of each line).

When you restart FrameMaker, this plugin will no longer be active.

To remove AutoFM from your FrameMaker installation:

- 1) Locate the “Pubs-Tools:AutoFM” and “Pubs-Tools:AutoFM-PDF” entries in the APIClients section of your *maker.ini* file in the main *FrameMaker* folder.
- 2) Delete these lines from the INI file, then save and close that file.
- 3) Delete the *autofm*. ** files from the *Pubs-Tools* folder located in the main *FrameMaker* folder (and possibly from your “AppData” folder if using Windows Vista, 7, or 8).
- 4) Delete the *AutoFM* folder from the *FrameMaker\Structure\xml* folder.
- 5) Remove the AutoFM entries from your structure application definitions file.

RELATED INFORMATION:

- “AutoFM” on page 1
- “The AutoFM Data Model” on page 7
- “Create and Run an AutoFM Script” on page 19
- “Troubleshooting” on page 9

3

The AutoFM Data Model

A simple data model that provides elements and structure for defining the opening, saving, processing, and printing of files in FrameMaker.

The AutoFM data model is very simple. Each element is named based on its function. Most elements have attributes that refine aspects of that functionality. In general an AutoFM script is an XML description of the process you'd go through if you were performing the task manually.

At the root is the autofm element, which can have the following child elements: file, book, action, or exit. The file and book elements let you open, select, close, print, or save (with like-named elements). The book element also lets you do an update to regenerate the book. The action element runs the specified FDK plugin, FrameScript, or ExtendScript (FM10 and later) command on the specified file. When processing is finished, the exit element closes FrameMaker.

RELATED INFORMATION:

"AutoFM" on page 1

"Installation and Setup" on page 3

"Create and Run an AutoFM Script" on page 19

Sample Script

This sample script shows how simple it is to automate processing in FrameMaker.

The following sample AutoFM script file is an example of a script that can be used to automate the creation of a FrameMaker book and PDF from a DITA map using DITA-FMx (the FMx-Auto “addon” is required to unlock the API in DITA-FMx). This is a description of the process defined by the XML script:

- `<file>` element opens the XML file named dita\sample.ditamap using the structure application named DITA-FMx-Map-1.2
- `<action>` element runs an FDK client named “ditafmx” and passes the specified arguments to that client (note that the syntax and format of the args element is defined by the client application)

- <book> element selects the book file that was just created by the FDK client, then saves it, then does a Save As PDF to sample.pdf, then closes the book file
- <file> element selects the original DITA map and closes it
- <exit> element closes FrameMaker (after closing any other files that may be open)

```
<autofm>
  <file>
    <open type = "XML">
      <name>dita\sample.ditamap</name>
      <appname>DITA-FMx-Map-1.2</appname>
    </open>
  </file>
  <action>
    <fdk>
      <client>ditafmx</client>
      <args>MapToBook | . | DITA-FMx-Book-1.2 | ..\book\sample.book</args>
    </fdk>
  </action>
  <book>
    <select>
      <name>book\sample.book</name>
    </select>
    <save/>
    <saveas type = "PDF">
      <name>sample.pdf</name>
    </saveas>
    <close/>
  </book>
  <file>
    <select>
      <name>dita\sample.ditamap</name>
    </select>
    <close/>
  </file>
  <exit/>
</autofm>
```

NOTE: *When generating a PDF, you can use the saveas or print elements. However, only the print element provides attributes that let you specify various document properties as well as enabling the Generate Acrobat Data option.*

RELATED INFORMATION:

"The AutoFM Data Model" on page 7

"AutoFM Elements" on page 10

"Create and Run an AutoFM Script" on page 19

Troubleshooting

Having trouble getting things working properly?

There are numerous reasons that the automation process may not work properly. We'll try to document those that come to mind and will add more as issues arise. If you run into trouble, contact us at <tools AT leximation DOT com>.

Remember that whenever anything unexpected happens, review the FrameMaker Console window for any messages or errors. If FrameMaker has closed, you can review the console messages for the last FrameMaker session by reading the *consfile.txt* file.

FrameMaker makes the “crash” sound when exiting

When FrameMaker exits (via the exit element in an AutoFM script), it may occasionally generate the “crash” sound. This appears to be caused by the operating system when it thinks the application has crashed (usually the case when an application closes itself). As long as the PDF has been created properly, and all files have closed successfully, this is not something to worry about.

On Windows 7, I’m getting the “Application has stopped responding” message

On a Windows 7 system, when FrameMaker exits, you may see the “Application has stopped responding” message which can interrupt automated scripting. This is generated by the operating system when it thinks the application has crashed, when in fact that was the intent. To prevent this dialog from displaying, make the following tweak to the registry:

```
HKEY_CURRENT_USER\Software\Microsoft\Windows\Windows Error Reporting\Dont Show UI = 1
```

FrameMaker doesn’t open the AutoFM XML file

The AutoFM structure application isn’t properly installed. Refer to step #2 in the “Installation and Setup” topic.

FrameMaker doesn’t automatically run the AutoFM script

Make sure that the AutoFM plugin is installed and registered. You should see two “AutoFM loaded” messages in the FrameMaker Console Window. Check the **Pubs-Tools > AutoFM** menu to make sure it’s available and registered.

Bookmarks are not created in the generated PDF

You need to use the “print” element and set the @acrobatdata attribute to “1” in order for bookmarks to be created.

Bookmarks are not set up properly

The default Job Options settings are used when a PDF is created. This value is defined in the current FM file or in the first FM file in a book. You must set this up ahead of time. A future update to AutoFM may allow you to specify this data in the script.

RELATED INFORMATION:

"AutoFM" on page 1

"Installation and Setup" on page 3

"The AutoFM Data Model" on page 7

"Sample Script" on page 7

"AutoFM Elements" on page 10

"Revision History" on page 21

AutoFM Elements

Name and description of the current AutoFM elements.

The following table shows the relationship of the top-level AutoFM elements and their child elements; it is not intended to be representative of a useful AutoFM script:

Table 3-1: Top-level AutoFM elements

action	book	file	exit
autofm	autofm	autofm	autofm
action	book	file	exit
extendscript	close	close	
script	export	export	
args	name	name	
fdk	import	import	
client	name	name	
args	template	template	
framescript	importformats	importformats	
script	name	name	
args	open	open	
select	name	name	
name	appname	appname	
	pdfsetup	pdfsetup	
	pdfdocinfo	pdfdocinfo	
	infoitem	infoitem	
	print	print	
	tofile	tofile	
	save	save	
	saveas	saveas	
	name	name	
	appname	appname	
	select	select	
	name	name	
	update		

action

Specifies the action to perform on books and files in FrameMaker. Allow you to select a file or book, then run a script or client command on that file. No attributes.

Contains: extendscript, fdk, framescript, select

Contained by: autofm

appname

Specifies the structure application name to use when opening or saving an XML or SGML file. No attributes.

Contains: *TEXT*

Contained by: open, saveas

args

Specifies the value of the argument passed to the FDK client or appended to the value of the ExtendScript or FrameScript script name. No attributes.

Contains: *TEXT*

Contained by: framescript, extendscript, fdk

autofm

The root element of an AutoFM file. No attributes.

Contains: action, book, file, exit

Contained by: *none*

book

Performs actions on books in FrameMaker. No attributes.

Contains: close, open, print, save, saveas, select, update

Contained by: autofm

client

Specifies the name of the FDK client. No attributes.

Contains: *TEXT*

Contained by: fdk

close

Closes a file in FrameMaker. No attributes.

Contains: *EMPTY*

Contained by: book, file

exit

Exits FrameMaker. Closes without saving all open documents before exiting. No attributes.

Contains: *EMPTY*

Contained by: autofm

export

Exports the current file from FrameMaker in the format defined by the filter attribute. The filter attribute specifies an “export filter hint string” (an 11 or 12 character string). The available export filter hint strings will vary based on the FM version and specific installation. To see the available hint strings, use the **Pubs-Tools > AutoFM > Filter Hint Strings** command.

Contains: name

Contained by: book, file

extendscript

Runs the ExtendScript specified by the script element. The value of the optional args element is appended to the script name. This is only available in FM10 and later. No attributes.

Contains: script, args

Contained by: action

fdk

Runs the specified FDK client with the value of the args element. No attributes.

Contains: client, args

Contained by: action

file

Performs actions on files in FrameMaker. No attributes.

Contains: close, open, print, save, saveas, select

Contained by: autofm

framescript

Runs the FrameScript specified by the script element. The value of the optional args element is appended to the script name. This is only available if the FrameScript plugin is installed. No attributes.

Contains: script, args

Contained by: action

import

Imports a file into FrameMaker of the format defined by the filter attribute. The filter attribute specifies an “import filter hint string” (an 11 or 12 character string). The available import filter hint strings will vary based on the FM version and specific installation. To see the available hint strings, use the **Pubs-Tools > AutoFM > Filter Hint Strings** command.

To apply an FM template to the file after importing, use the template child element.

Contains: name, template

Contained by: book, file

importformats

Imports the specified formats from the specified file (name node) into the current file (defined by the parent file or book node). Attributes are available for each option in the **Import Formats** dialog. All attributes default to “0” (disabled); enable (set to “1”) those that you want to import (or use the “alloptions” attribute). Seventeen attributes (depending on FM version):

- **alloptions** - Enables all options (FM10 and later, overrides other settings).
- **paragraphfmts** - Sets the paragraph formats option.
- **characterfmts** - Sets the character formats option.
- **pagelayouts** - Sets the page layouts option.
- **tablefmts** - Sets the table formats option.
- **colordefs** - Sets the color definitions option.
- **documentprops** - Sets the document properties option.
- **refpages** - Sets the reference pages option.
- **variabledefs** - Sets the variable definitions option.
- **crossreffmts** - Sets the cross-reference formats option.
- **contextsetgs** - Sets the conditional text settings option.
- **mathdefs** - Sets the math definitions option.
- **combinedfonts** - Sets the combined fonts option (FM9 and later).
- **fbasettings** - Enables the paragraph formats option (FM10 and later).
- **objectstyles** - Enables the paragraph formats option (FM12 and later, even though this was available in FM11; use “alloptions” to enable in FM11).
- **removebreaks** - Enables the paragraph formats option.
- **removeexceptions** - Enables the paragraph formats option.

Contains: name

Contained by: book, file

infoitem

Defines a PDF Doc Info entry (field/value pair). Add one infoitem element for each entry. One attribute (required).

- **infofield** - Specifies the field name of the PDF Doc Info. Known values are Author, Title, Subject, Keywords, Copyright, Web Statement, Job, Marked. No default.

Contains: *TEXT*

Contained by: pdfdocinfo

name

Specifies the name of the file to export, import, open, save, or select. No attributes.

Contains: *TEXT*

Contained by: export, import, select, open, saveas

open

Opens a file in FrameMaker. One attribute:

- **type** - Specifies the type of file to open. Values are FM, MIF, XML, SGML. Default is FM.

Contains: name, appname

Contained by: book, file

pdfdocinfo

Defines the PDF Doc Info entries. No attributes.

Contains: infoitem

Contained by: pdfsetup

pdfsetup

Assigns a PDF Job options name. Three attributes:

- **pdfjoboptions** - Specifies the PDF joboptions name.
- **pageheight** - Specifies the height of the generated PDF (use standard units suffix string, such as “in”, “cm”, “mm”, etc.).
- **pagewidth** - Specifies the width of the generated PDF (use standard units suffix string, such as “in”, “cm”, “mm”, etc.).

Contains: pdfdocinfo

Contained by: book, file

print

Prints a file in FrameMaker. Use the `tofile` element to specify the name of the print file. Attributes match the options in the FrameMaker Print dialog:

- **printername** - Specifies the name of the printer. No default value.
- **prinrange** - Indicates the print range. Values are “all” or “range”. Default value is “all”.
- **startrange** - If the value of `prinrange` is “range”, this value sets the start page range number. No default value.
- **endrange** - If the value of `prinrange` is “range”, this value sets the end page range number. No default value.
- **oddpages** - If set to 1, enables the “Odd-Numbered Pages” option. Default value is “1” (On).
- **evenpages** - On set to 1, enables the “Even-Numbered Pages” option. Default value is “1” (Off).
- **collate** - If set to 1, enables the “Collate” option. Default value is “1” (On).
- **lastsheetfirst** - If set to 1, enables the “Last Sheet First” option. Default value is “0” (Off).
- **skipblankpages** - If set to 1, enables the “Skip Blank Pages” option. Default value is “0” (Off).
- **spotcolorasbw** - If set to 1, enables the “Spot Color as Black/White” option. Default value is “0” (Off).
- **lowresimages** - If set to 1, enables the “Low-Resolution Images” option. Default value is “0” (Off).
- **regmarks** - Specifies the value for the “Registration Marks” option. Valid values are “None”, “Western”, and “Tombo”. Default value is “None”.
- **thumbnails** - If set to 1, enables the “Thumbnails” option. Default value is “0” (Off).
- **thumbrows** - Specifies the number of thumbnail rows if the `@thumbnails` is set to “1”. No default value.
- **thumbcols** - Specifies the number of thumbnail columns if the `@thumbnails` is set to “1”. No default value.
- **copies** - Specifies the number of copies to be printed. Default value is “1”.

- **scale** - Specifies the scale of the print job. Default value is “100%”.
- **acrobatdata** - If set to 1, enables the “Generate Acrobat Data” option. Default value is “0” (Off).

Contains: tofile

Contained by: book, file

save

Saves the current file or book. No attributes.

Contains: *EMPTY*

Contained by: book, file

saveas

Performs a Save As operation on the current file or book. One attribute:

- **type** - Specifies the Save As operation to perform. Standard values are FM, MIF, XML, SGML, TXT, and PDF. Additional version-specific values are FM6, FM7, FM8, FM9, FM10, FM11, FM12, MIF7, MIF8, MIF9, MIF10, MIF11, MIF12, COMPOSITE, BOOK-XML, BOOK-FM. Default is FM.

Contains: name, appname

Contained by: book, file

script

Specifies the name of an ExtendScript or FrameScript. No attributes.

Contains: *TEXT*

Contained by: framescript, extendscript

select

Selects a file in FrameMaker for processing. No attributes.

Contains: name

Contained by: action, book, file

template

Specifies the name of the file to use as the template for the file imported with the import command. No attributes.

Contains: *TEXT*

Contained by: import

tofile

Specifies the print file name to create. No attributes.

Contains: *TEXT*

Contained by: print

update

Updates the current book. No attributes.

Contains: *EMPTY*

Contained by: book

RELATED INFORMATION:

"The AutoFM Data Model" on page 7

"Sample Script" on page 7

"Create and Run an AutoFM Script" on page 19

4

Create and Run an AutoFM Script

AutoFM scripts are simple XML files that can be created in FrameMaker or by other tools and processes.

AutoFM scripts must have a suffix of “autofm.xml”. You can provide additional text as a prefix. For example, *mybook-autofm.xml*. The following describes how to create an AutoFM script.

- 1) Choose **Pubs-Tools > AutoFM > New**.

You will be prompted for a file name, then a new AutoFM XML file is created.

- 2) Add and modify elements and attributes to create the process needed.

The Element Catalog shows the valid elements. Check the attributes for each element to see if there are any properties that should be set for elements you’ve added. Be sure to add the exit element to close FrameMaker after processing is complete.

- 3) Always test a new script manually by choosing **Pubs-Tools > AutoFM > Run** command.

Once you’ve confirmed that everything is working as expected, proceed to automated testing using an external scripting environment. Set up a batch file or other command line scripting tool to run your AutoFM script.

Any scripting language that can run an executable should be able to drive your AutoFM script (DOS batch script, Perl, Ant, etc.). You just specify the FrameMaker executable and pass the AutoFM script to it as a command line parameter. It is typically best to specify the full path to both the FrameMaker executable and the AutoFM script, but this can change depending on your environment. Be sure to wrap paths in quotes if they contain spaces.

To run the AutoFM script just run the FrameMaker executable and pass the AutoFM filename as an argument, as in the following example:

```
"C:\Program Files (x86)\Adobe\AdobeFrameMaker12\Framemaker.exe"  
"C:\Project\mybook-autofm.xml"
```

Another way to run the AutoFM script is to right-click the AutoFM file in Windows Explorer, and choose the Open With command then select FrameMaker.

If the AutoFM plugin is installed properly (both the DLL and the structure application), when FrameMaker starts and opens the AutoFM file, it begins processing the instructions in that file. The results of each step are saved to an *<autofm-file>.log* file. For additional debugging, the *consfile.txt* file (which contains the messages from the Console Window), is copied to a *<autofm-file>_consfile.txt* file in the same folder as the AutoFM file.

To generate a PDF you can use the saveas or print elements. At this time, in order to specify the various PDF properties you must use the print element and its related attributes. When printing a file to PDF, the default PDF Job Options file associated with the FM or BOOK file is used.

TIP: *When using FM9 or later, it's best to disable all pods as well as the Structure View window.*

RELATED INFORMATION:

- "AutoFM" on page 1
- "Installation and Setup" on page 3
- "The AutoFM Data Model" on page 7
- "Sample Script" on page 7
- "AutoFM Elements" on page 10
- "Troubleshooting" on page 9

A

Revision History

Describes the changes between versions of AutoFM.

RELATED INFORMATION:

"AutoFM" on page 1

"Installation and Setup" on page 3

"The AutoFM Data Model" on page 7

"Troubleshooting" on page 9

1.0.05 - 5 October 2015

New Features

Added support for FrameMaker 2015

AutoFM can now be installed on FrameMaker 2015.

Added pdfdocinfo element

The pdfdocinfo and infoitem elements let you define PDF Doc Info meta-data entries.

Added pageheight and pagewidth attributes

The pdfsetup element can now specify pageheight and pagewidth values.

Structure Application Updates

DTD and EDD (autofm.dtd, autofm-edd.fm)

- Added pdfdocinfo and infoitem elements, and related attributes.
- Added pageheight and pagewidth attributes to the pdfsetup element.
- Added FM version-specific @type values for the saveas element (FM12 and MIF12).

Template (autofm-template.fm)

- Imported updated EDD.

Bug Fixes / Minor Updates

NONE

1.0.04 - 20 April 2014

New Features

Added support for FrameMaker 12

AutoFM can now be installed on FrameMaker 12.

Added pdfsetup element

The pdfsetup element allows you to set the joboptions value on a document.

Added importformats elements

The importformats element imports formats from one document to another.

Structure Application Updates

DTD and EDD (autofm.dtd, autofm-edd.fm)

- Added pdfsetup and importformats elements, and related attributes.
- Added FM version-specific @type values for the saveas element (FM11 and MIF11).

Bug Fixes / Minor Updates

Modified the method for exiting FrameMaker

The standard method for exiting FrameMaker from an API client seems to have problems on Win7 and 8 systems. This plugin now uses the

PostQuitMessage() WinAPI function, which seems to be a bit more graceful.

1.0.03- 19 August 2012

New Features

NONE.

Structure Application Updates

NONE.

Bug Fixes / Minor Updates

Minor update.

1.0.02- 27 July 2012

New Features

Added support for FrameMaker 11

AutoFM can now be installed on FrameMaker 11.

Added support for deleting PrinterJobControl entries

If you're generating a lot of PDFs with AutoFM, you may not want each PDF name stored in the registry. The General/DeletePdfRegEntry parameter in the *autofm.ini* file specifies if these entries are stored or deleted.

Structure Application Updates

NONE.

Bug Fixes / Minor Updates

Properly allows printing to PDF on 64-bit systems

Fixed a bug that was forcing the system to prompt for the PDF filename on 64-bit systems.

1.0.01- 7 July 2012

New Features

Added import and export elements

New import and export elements allow the specification of an import or export filter via the filter attribute. The import element also provides for a template child element that lets you specify a FM file template to apply to the imported file.

Added Filter Hint Strings command

The Filter Hint Strings command generates a list of the installed import and export filters. The strings displayed can be used with the import and export filter attribute.

Added support for additional saveas file types

The saveas element now accepts additional @type values. The available values are not valid for all versions of FrameMaker; use those that make sense for your FrameMaker version.

Structure Application Updates

DTD and EDD (autofm.dtd, autofm-edd.fm)

- Added import, export, and template elements.

- Added FM version-specific @type values for the saveas element (FM6, FM7, FM8, FM9, FM10, MIF7, MIF8, MIF9, MIF10, COMPOSITE, BOOK-XML, BOOK-FM).

Bug Fixes / Minor Updates

Console file is copied to the “log” folder

After an AutoFM build completes, the *console.txt* file is copied to the folder that contains the generated LOG file (same folder as the AutoFM file).

Properly generates PDFs on 64-bit Windows systems

PDF printing on 64-bit Windows systems is handled by the *splwow64.exe* utility. If this file is not installed in the standard location (*C:\Windows\splwow64.exe*), specify its location in the General/SPLWOW64 parameter in the *autofm.ini* file.

1.0.00 - 4 February 2012

Initial release.

Index

A

action, perform on file or book	11
AutoFM elements	
action	11
appname	11
args	12
autofm	12
book	12
client	12
close	12
exit	12
export	12
extendscript	13
fdk	13
file	13
framescript	13
import	13
importformats	14
infoitem	14
name	15
open	15
pdfdocinfo	15
pdfsetup	15
print	15
save	17
saveas	17
script	17
select	17

template	17
tofile	17
update	17
AutoFM sample	
action element	7
book element	7
exit element	7
file element	7
AutoFM script model	7
AutoFM structure application	3
autofm.ini file	4

C

close file or book	12
create script	19

D

data model	7
DeletePdfRegEntry INI parameter	4

E

exit FrameMaker	12
export	12
ExtendScript, run on file or book	13

F

FDK client, run on file or book	13
FrameMaker versions, supported	1

FrameScript, run on file or book	13	print to PDF	15
I		PrinterJobC ontrol registry entries	4
import formats	14	R	
import, into file	13	run script	19
installing AutoFM	3	S	
J		sample AutoFM script	7
joboptions	15	save file or book	17
O		SPLWOW64 INI parameter	4
open file or book	15	splwow64.exe	4
P		structure application	11
PDF joboptions	15	structure application installation	3
plugin installation	3	U	
print file or book	15	uninstalling AutoFM	5
		update book	17