



Features

Tips

Reviews

Case Studies

Adobe Speaks



10 Neat Things You Can Do with the Maker.ini File

by Dave Valiulis

[EMAIL](#)

The maker.ini file is the initialization file for FrameMaker in Windows. This file is a simple text file that FrameMaker looks at when starting up. From this file, it reads in a bunch of settings and properties. The cool thing is that you can edit this file yourself to affect FrameMaker's settings in some nifty ways.

But before we get started, some preliminary info is in order:

- The info in this article applies only to the Windows version of FrameMaker (not MacOS or Unix, and not FrameMaker+SGML or FrameViewer). Moreover, some of the options mentioned here apply only to version 5.5.6.
- The maker.ini file is in the main program folder, typically in a path that looks like this:
C:\Program Files\Adobe\Framemaker5.5\maker.ini.
- Be sure to make a copy of the original maker.ini file before you begin making changes. You may have to revert to the original.
- If you use FrameMaker itself to edit the file, be sure to save the file as text, not as a FrameMaker file.
- You'll usually have to exit and then restart FrameMaker to have your changes take effect.
- Refer to chapter 2 of the online manual "Customizing FrameMaker Products" for fuller instructions on working with the maker.ini file.

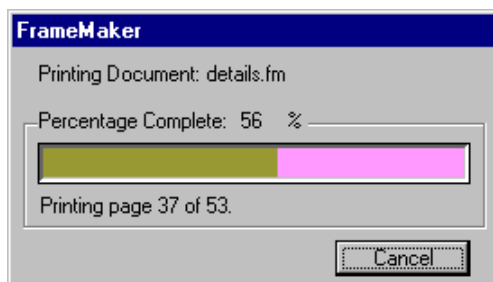
There are hundreds of settings in the maker.ini file, but most of them are pretty boring in that they let you muck with settings you would never really need to change or they duplicate settings that are more easily accessed from within the program.

Useless things

An example of the former case is being able to change the colors used in progress bars, which the maker.ini file calls "thermometers." The most obvious progress bar in the program is the red one that shows up when FrameMaker is first launching. But guess what? Changing these settings doesn't affect that progress bar. It does, however, affect the one that is displayed when you print a document.

For example, changing the default RGB color values for the thermometer to the ones shown results in the gaudy thermometer shown in the next figure.

```
ThermoDoneColor=153, 153, 51
ThermoRemainingColor=255, 153, 255
```



And speaking of wonderfully useless options, how about being able to change the font used in the start-up screen (which the ini file calls the "banner window")? Yup, you can get rid of boring old Arial and use, say, ZapfDingbats!

```
BannerFont=ZapfDingbats
```



Redundant things

There are also a host of things you can do in the ini file that are completely redundant with things you can do in FrameMaker itself - and do them much easier in FrameMaker. Who knows why, for example, the architects of FrameMaker thought it necessary to give you control over where a window appears when it's first opened? This is especially odd since FrameMaker windows remember their location so that it's infinitely easier to position the Find/Change dialog in the lower left corner of the window than to enter this in the ini file:

```
Search=14, 565
```

Lots of other options are settable in the ini file that are more intuitively

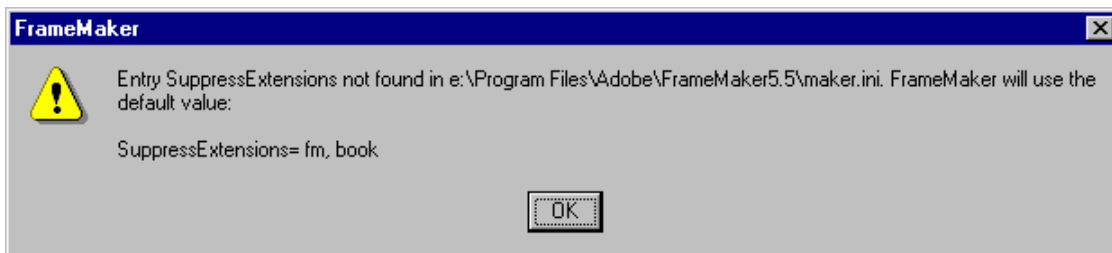
done from within the product, such as spell-check settings, doc-compare settings, and preferences. There's just no need to linger at these.

Dangerous things

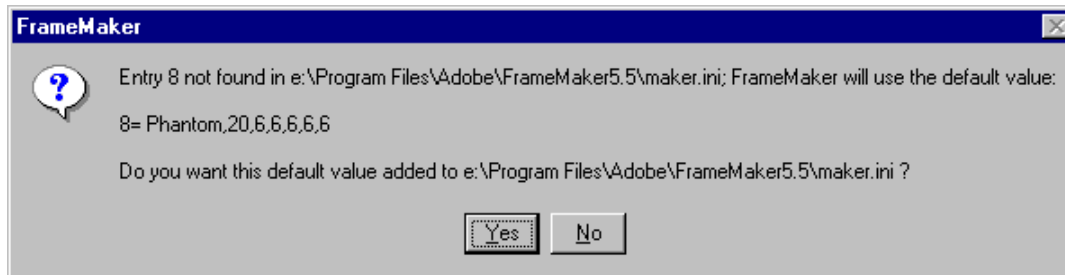
Among the many options you can muck with, there are also a handful you should probably never tinker with. For example, steer clear of the ones that describe PostScript, imaging, and API.

```
UsePostscript=On  
GDIRasterizeKiloPixelLimit=8192  
API=On
```

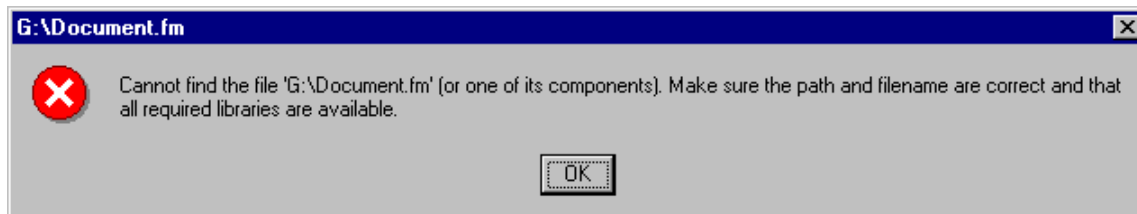
The good news is that if you accidentally enter an impossible value for an option, FrameMaker can often gracefully recover, as the following alert message shows:



Sometimes, FrameMaker will even volunteer to reinstate a missing line, such as the following:



But to play it safe, make a backup of the ini file before you begin. Otherwise, you might be stuck with the following less-than-helpful alert, which appears when the maker.ini file can't be found or opened:



Neat things

Now to the fun stuff. Nestled in the ini file innards are a handful of useful, albeit obscure, features. These are things that can't be done from within the product or can't be done as easily. Let's start the countdown for the Top 10 Things You Can Do in the Ini File:

#1: Use your templates efficiently

Two options in the ini file let you take better advantage of your custom or inhouse template. The first looks like this:

```
TemplateDir=templates
```

This setting lets you specify a folder that holds the templates you want to use. This is the folder that FrameMaker will point to by default when you choose File > New. If your workgroup uses a special template set, it's a nice convenience to have this folder opened for you whenever you want to create a new document.

The location can be a network drive. So, for example, you could modify the setting to:

```
TemplateDir="F:\public\frame\new templates\"
```

(Be sure to use quotes around a path with a space in it.) This strategy gets everyone in the workgroup to point to a single location. That way, when the template is updated, everyone will automatically start using the new version.

Another template-related option is new as of version 5.5.6 and is a bit more esoteric:

```
UseDisplayUnitInTemplate=Off
```

Off is the default for this setting, but if you set UseDisplayUnitInTemplate to On, the default display units for the ruler, grid, and snap are taken from the document. When set to Off, the values are taken from what in the ini file itself. This makes a difference because you may want your template users to be always working in points, but unless you set this to On, no amount of tweaking the template to only use points will assure that the users will only see points.

#2: Turning Snap off, sort of

Were you hoping I was going to tell you that there's an ini setting for turning snap off? Well, dream on. The default setting for Snap is On, and there's no way to change the default to Off in Windows. (It's infuriating that you can do this in both the MacOS and Unix versions of FrameMaker, but not in Windows.) If you choose Graphics > Snap, that will turn snap off while only off for that session. The next time you start FrameMaker, snap will be back on. Ugh.

But there is some good news. You can modify two settings in the ini file so that a new file - by default - uses an incredibly small snap, which in effect turns it off.

Look for these lines:

```
DefaultSnapInch=0.125in  
DefaultSnapCm=0.25cm
```

Now change them to these:

```
DefaultSnapInch=0.0001in  
DefaultSnapCm=0.0001cm
```

Unfortunately, these settings apply only to new documents. Existing documents will still have the Snap values they always did. But if you change the Snap values in your templates and change the ini file as described here, eventually snap will be off disabled in all your documents. Another downside of this hack is that if you ever really do want a Snap grid on, there's no easy way to enable it. (You can do it only by choosing View > Options and changing the 0.0001 value to something bigger such as 0.1.) But if you're like me, you never need Snap anyway, so good riddance to bad garbage!

By the way, the unit used by FrameMaker for snapping (in or cm) is not specified from within FrameMaker at all. It's governed by the "Measurement System" option in the Windows Regional Settings control panel. That's why it's a good idea to set both of the ini options at the same time.

#3: If you find autofit awful...

I have to admit that I'm embarrassed by this one. The autofit feature in FrameMaker 5.5 and later was implemented on my recommendation. I was getting tired of FrameMaker wasting so much screen real estate on material that never changed and was rarely in need of editing - namely, the master-page header/footer area. I said wouldn't it be great if FrameMaker could automatically show you just the text frame. The engineers agreed, and the Fit Window to Text Frame option in the zoom pop-up menu was born.

But this feature had a side-effect that a lot of people really disliked. After you've chosen Fit Page in Window or Fit Window to Page from the Zoom pop-up menu, the window size changes when you change the zoom level. So what? Well, it turns out that a lot of people like to click the little z or the big Z several times in a row to zoom incrementally. And they couldn't do that anymore because the z's moved around when the window resized. Hence, the DisableAutofitAfterZoom option was born.

```
DisableAutofitAfterZoom=Off
```

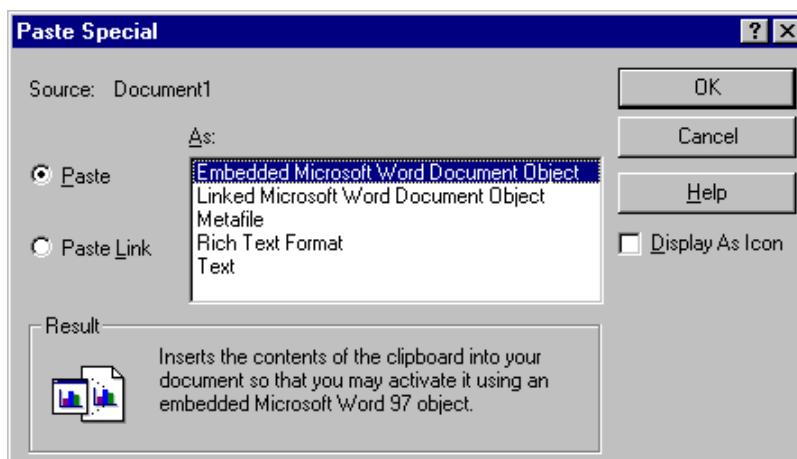
If you want to be able to have the little z's stay in one place, change this setting to On. The only way to do this from within the program is to use a little-known Esc-key shortcut: Esc z a f

#4: When pasting produces the unexpected

Users are often mystified when they copy text from Word and paste it into FrameMaker. The result is not what they expected or wanted, for what they get is an embedded OLE object (so that double-clicking it

reopens it in Word!). All they wanted was the text pasted, not anything else.

Savvy users know they can override this OLE orneriness by right-clicking and choosing Paste Special. Then, they can choose how the text should be pasted (as text, as an embedded object, etc.). The following figure shows that there are actually five types of data contained in the Clipboard when you copy text from Word.



But most users would rather avoid having to go into a dialog box simply to paste a paragraph from Word. Help is on the way. Look for this line in the ini file:

```
ClipboardFormatsPriorities=
ILE,OLEE2,EMF,META,DIB,BMP,MIF,RTF,TEXT
```

This line specifies the order for the formats used to paste Clipboard items - from left (highest) to right (lowest). Looking at the list reveals the problem: OLE has a higher priority than TEXT (that is, it's to the left of TEXT). This means if the Clipboard contents could be pasted as either an OLE object or as plain text, it will be pasted as an OLE object. If you're like me and almost never want an OLE object pasted, change the line so that OLE is last:

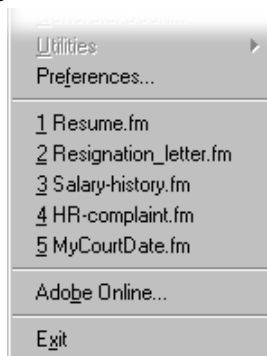
```
ClipboardFormatsPriorities=
FILE,EMF,META,DIB,BMP,MIF,RTF,TEXT,OLE 2
```

And if you really want to make sure text is pasted as text, move TEXT so that it's first:

```
ClipboardFormatsPriorities=
TEXT,FILE,EMF,META,DIB,BMP,MIF,RTF,OLE 2
```

#5: Covering your tracks

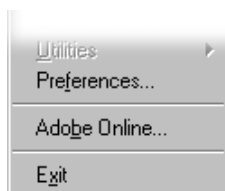
FrameMaker lists the last five most recently opened files at the bottom of the File menu. This is nice because it offers a quick way to reopen a file that you've been working on. But what if you've had a really bad week, and the last five files you've been working on look like this:



If you think it's possible that someone else may be using your copy of FrameMaker, you may want to purge this file list. You can do it the hard way by systematically opening five documents whose filenames would be less incriminating. Or you can do it more efficiently by simply removing these lines, which would appear immediately below the [RecentlyVisitedFiles] header:

```
0=C:\Resume.fm
1=C:\Resignation_letter.fm
2=F:\Salary-history.fm
3=F:\HR-complaint.fm
4=F:\MyCourtDate.fm
```

By removing these five lines, you purge the file list, and the tabula rasa result is as follows:



#6: Dash it all!

Not many people use dashed lines in FrameMaker, but if you do, did you know you could define your own dash patterns? FrameMaker offers eight predefined patterns that are pretty standard. Using lines in the ini file, you can change any of the eight to ones you define by using numbers representing the line lengths and gaps (in points). Here are eight new ones I defined:

```
1=Grow, 2, 6, 6, 6, 10, 6, 14, 6
2=Shrink, 14, 6, 10, 6, 6, 6, 2, 6
3=Victory, 2,4,2,4,2,4,10,4
4=Long-short, 20, 4, 4, 4
5=Not Faint, 4,4
6=Faint, 3, 3
7=Fainter, 2, 2
8=Faintest, 1,1
```

And here is how they look in the Dashed Line Options dialog in FrameMaker:



#7: If Size matters, make it bigger

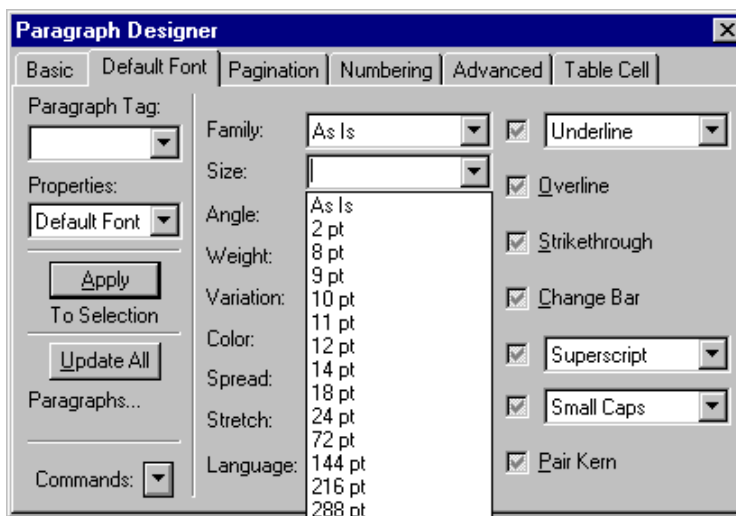
FrameMaker lets you work with type in sizes from 2 to 400 points. That's quite a range, but it is only hinted at by the predefined sizes in the Size pop-up menu in the Paragraph and Character Designers. There, the 10 values ranging only from 7 to 36 pt. If you want to add more values to this pop-up or change the ones displayed, monkey around with the Sizes line in the ini file. The default line looks like this:

Sizes=7pt, 8pt, 9pt, 10pt, 11pt, 12pt, 14pt, 18pt, 24pt, 36pt

Let's change it to this:

Sizes=2pt, 8pt, 9pt, 10pt, 11pt, 12pt, 14pt, 18pt, 24pt, 1in, 2in, 3in, 4in

Now take a gander at the Size pop-up, which has grown from 10 values to 13:



Even though the Size pop-up always displays values in points, FrameMaker lets you enter sizes in other units (such as inches) in the ini file; it then does the math for you and displays what the point-equivalent is.

Also, this technique does not change the contents of the Format > Size menu. To change that, you have to fiddle with the menu config files, not the ini file.

#8: When a font can't be found...

Users who hop from platform to platform know that FrameMaker does a great job of keeping your documents usable on all platforms but the one irksome area is fonts. One of the major font hassles is that the identical font on different platforms is not always recognized as being identical due to differing naming conventions on each platform. In other words, the font is there, but FrameMaker doesn't recognize that it's there due to some subtle naming difference. The ever-accommodating ini file tries to come to the rescue in this vexing area by offering font mapping entries that you can enter so that the fonts will be recognized successfully.

Unfortunately, the syntax for these mappings is obscure and finicky, which means most users don't bother with font mappings and rely on the Remember Missing Fonts preference option (and thank goodness for that option!).

Still, font mappings can be a big convenience once they're set up correctly in the ini file. The best way to make the mappings right, however, is not to type them in yourself. Instead, you'll want to copy them from a handy document that Adobe provides on their web site:

<http://www.adobe.com/support/techdocs/112da.htm>

This document has the mappings for the following common fonts, which you can copy and paste for perfect results:

- ACaslon
- Agaramond
- Frutiger
- Futura
- Garamond
- Helvetica Condensed
- Helvetica Narrow
- Letter Gothic
- Optima
- Stone Sans
- Stone Serif
- Univers
- Utopia

But sometimes a font can't be found simply because it's not there. Never has been. Never will be. What then?

By default, FrameMaker will substitute a font for it, such as Times or Helvetica, and reformat the document with that new font. But if you're using Adobe Type Manager 4.0 or later, you can perform a more

sophisticated substitution trick.

You can have ATM synthesize a font for the missing one that exactly matches the font metrics of the missing one. This results in a new look for the font, but the line breaks and page lengths will remain unchanged. If this is important to you, this is a great feature. All you do is change this ini line

```
AtmFontSubstitution=Off
```

to this:

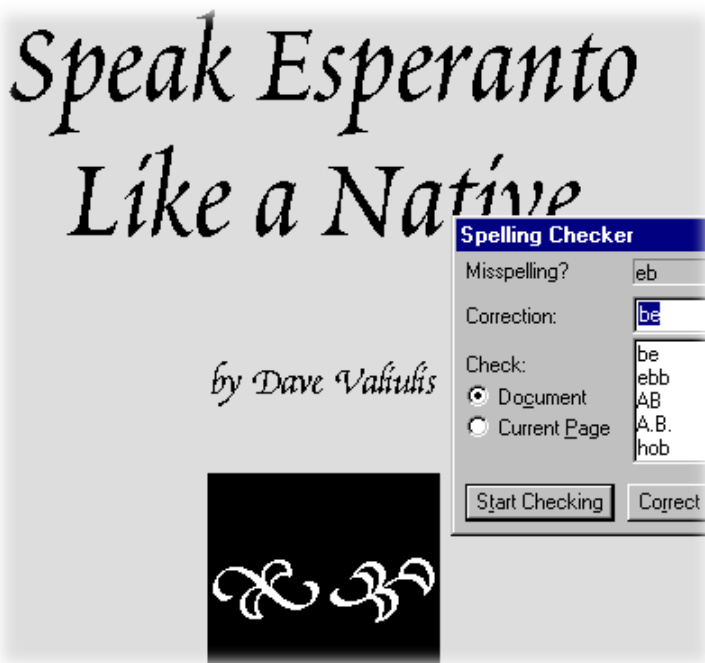
```
AtmFontSubstitution=On
```

However, be warned that ATM font substitution can be slow, especially if the font is used a lot. Still, with the super fast systems available nowadays, you may want to turn it on and see if you like the results.

#9: When a font ain't text

Another font trick you can do in the ini file is to tell FrameMaker which fonts aren't text. By informing FrameMaker of this, you save yourself the hassle of trying to spell-check such fonts. Nontext fonts include dingbat fonts, pi fonts, icon fonts, and the like.

For example, the following page uses a special ornament font to match the Poetica typeface. By typing the letters *e* and *b* in this font, the graceful flourishes appear instead. Sadly, when spell-checked, FrameMaker selects the characters and flags them as misspelled.



FrameMaker starts off knowing about a number of nontext fonts, but this isn't one of them. Here's the entry with the Poetica Ornaments added at the end of the line:

```
NonTextFamilies=ZapfDingbats,Symbol,WingDings,
Monotype Sorts,Euro Monospace, Euro Sans,Euro
Serif,Poetica SuppOrnaments
```

Result? Now when you spell-check a document containing these ornaments, you're far more likely to see this:



#10: Quoting in tongues

Most users like the Smart Quotes feature in FrameMaker, which automatically uses curly quote marks when you press the double-quote key ("). But did you know you can change what characters are used for these curly quotes? Look for the SmartQuotes line in the ini file. There, you'll find the hex codes for the quote marks used in several languages. Copy and paste them into the SmartQuotes line.

For example, pasting the French codes makes the line look like this:

```
SmartQuotes=\xdc\xdd\xc7\xc8
```

Now, when it's time to use curly quotes, they come out using the appropriate «guillemets» character, as shown. Also available are the characters commonly used in German and Swedish/Finnish.

English	“double” and ‘single’ quotes.
German	„double“ and ‚single‘ quotes.
French	«double» and «single» quotes.
Swedish/Finnish	”double” and ’single’ quotes.

And if you're feeling particularly iconoclastic, you could enter (the hex code for) any characters, and they will be used in the place of the curvy quotes.

Have fun!

These are my top 10 neat things to change in the maker.ini file, but feel free to go exploring on your own. (You did make a backup, didn't you?) You may find some real gems in there that I overlooked. Have fun.

