NAME

style.conf – GMA GUI Style Configuration File

DESCRIPTION

This file is now deprecated. Its function is now superseded by the in-app preference editor.

The `gma-mapper`(6) battle grid program, along with other GUI tools such as `gma-initiative-clock`(6), use colors and fonts to indicate a number of things on their displays. The `style.conf` file described here allows the user to customize any or all of these settings. This file may be specified to the application using command-line options. Otherwise, if the file `~/.gma/mapper/style.conf` exists, it will be read by default. See the manpage entries for the individual programs for details.

The overall format of this file is that of a classic Windows INI file; that is, the file is divided into named stanzas. Within each stanza is a set of named values, one per line.

Each stanza begins with a line containing only the stanza name in square brackets. Each value is defined by a line that begins with the value name, followed by an equals sign, followed by the value itself.

Tool Stanzas

A stanza with the name of a GMA tool defines the style settings for that tool. Currently-supported tool names (i.e., the names of these stanzas) include just "`mapper`". The tool-name stanza may include the following definitions:

`fonts=`*stanza* Specifies the stanza which defines the custom fonts to be used by this program.

`dierolls=`*stanza*
Specifies the stanza which defines the custom styling to be used for die roll displays.

Font Definitions

A font definition stanza defines named fonts with lines of the form

fontname=definition

The definition of the font is provided by giving option-value pairs. You need only provide the options which differ from the default for a given font for your system. If the value in any of these pairs contains spaces, it must be enclosed in curly braces. Options and values are separated from each other with whitespace.

The available options are:

`-family` The name of the font. The underlying Tk system provides the family names `Courier`, `Helvetica`, and `Times`, which are representative names for a fixed-width, sans serif, and serifed font, respectively; these may not necessarily be fonts with those exact names. You may also use the name of any font known on your system. This name is case-insensitive.

`-size` The size in points of the font, if a positive value. If negative, its absolute value specifies the font size in pixels.

`-weight` One of `normal` or `bold`.

`-slant` One of `roman` or `italic`.

`-underline`
A boolean value; if true, characters drawn in this font will be underlined.

`-overstrike`
A boolean value; if true, characters drawn in this font will have a line struck through their centers. Note that there is a style setting (q.v.) which will specify overstriking for a specific display type. This is generally preferred to setting it font-wide so the same font may be used for values which are overstruck and for others which are not.

Example:

```
[mapper]
fonts=myfonts
[myfonts]
hack=-family Hack -size 12
fancy=-family {Fancy Script Regular} -size 24 -slant italic
```

Die Roll Style Stanzas

There are a number of die roll reporting styles in effect to display the various elements that make up a report of a die roll result. The color and font of each may be controlled by entries in this stanza. Any not provided in this stanza will assume their built-in default values.

`bg_`*element*`=`*color* Sets the color drawn behind text for the die-roll element named.

`bg_list_even=`*color* Sets the background color for even-numbered entries in the recent and preset die roll lists.

`bg_list_odd=`*color* Sets the background color for odd-numbered entries in the recent and preset die roll lists.

`default_font=`*name* Use the user-defined font *name* for all element fonts where no explicit `font_`*element* assignment exists in this file.

`fg_`*element*`=`*color* Sets the color of the text for the die-roll element named.

`fmt_`*element*`=`*string* Specifies the formatting to be used for the reported *element*. By default, some descriptive punctuation, spacing, and text are presented around the variable values being reported. If these are not how you would like those elements to be shown, specify your own string in this configuration value. The *string* is a `printf`(3)-style format string, with a "`%s`" at the location you would like the data item(s) to appear.

Since whitespace is allowed around the keys and values in an INI file, which is automatically ignored, a mechanism is provided to allow your format string to explicitly contain leading and/or trailing spaces. If your format string (excluding any leading or trailing spaces) begins or ends with a vertical bar ("|"), the bar(s) mark the explicit end(s) of the string value. Thus, given the configuration file lines:

```
fmt_a=  (%s)
fmt_b=| (%s) |
fmt_c=| %s
fmt_d= %s |
```

the resulting format string for element `a` will be "`(%s)`" (with all spaces stripped); the format for `b` will have a space before and after the `(%s)` part; `c` will have a space before its content; and `d` will have a space after its content.

`font_`*element*`=`*name* Sets the font used for the die-roll element named. This font name must be defined in the selected font definition stanza in this file.

`offset_`*element*`=`*delta* Display the text of this element *delta* pixels higher than the baseline of the surrounding text. The *delta* value may be negative to lower the text.

`overstrike_`*element*`=`*bool* A boolean value; if true, text for this element will have a line struck through it.

`underline_`*element*`=`*bool* A boolean value; if true, text for this element will be underlined.

The *element* names may be `best`, `bonus`, `constant`, `critlabel`, `critspec`, `dc`, `diebonus`, `diespec`, `discarded`, `exceeded`, `fail`, `fullmax`, `iteration`, `label`, `max`, `maximized`, `maxroll`, `met`, `min`, `moddelim`, `operator`, `repeat`, `result`, `roll`, `separator`, `short`, `subtitle`, `success`, `until`, or `worst`, the meanings of which are described in `gma-dice`(3). In addition, the elements `fullresult` and `title` control the display of the overall total and user-assigned title for the die roll, `comment` describes comments about the die roll from the server, `system` is for system-generated messages, and the elements `from`, `normal`, and `to` control the appearance of chat messages generally.

A *color* value may be the name of a color as known to the system (e.g., listed in the X11 *rgb.txt* file), such as "`alice blue`" or "`PaleGreen4`", or a hexadecimal color value in the format #*rgb*, #*rrggbb*,

`#`*rrrgggbbb*, or `#`*rrrrggggbbbb*. System-defined names such as "`systemActiveAreaFill`" (macOS) and "`system3dLight`" (Windows) are also recognized where available. The X11 names are typically implemented by the underlying frameworks even on non-Unix-like systems so they are generally usable everywhere.

Example:

```
; This is an example style.conf file which
; shows how to customize the mapper tool.
[mapper]
dierolls=mydice
fonts=myfonts

[mydice]
fg_fullresult=red
font_fullresult=hack
fmt_roll=(rolled {%s})

[myfonts]
hack=-family Hack -size 12 -weight bold
```

HISTORY

This feature appeared in version 3.33 of the mapper tool.

AUTHOR

Steve Willoughby / steve@madscience.zone.

SEE ALSO

`gma-dice`(3).

BUGS

The Tcl library support for reading INI files isn't as enhanced as the one in the Python library, so since the mapper uses this configuration file it needs to conform to the simpler rules supported by the Tcl library: no multi-line values, keys and values separated only by an equals sign, and only supports "`;`" as the comment character.

Since this is (eventually) intended to apply to more tools than the mapper, this really should have been located in `~/.gma` instead of `~/.gma/mapper`. A future version will either move the default location or search both directories.

COPYRIGHT