Macro

Ron Goertz

COLLABORATORS

	TITLE : Macro		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY	Ron Goertz	February 12, 2023	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

Macro

1.1 INDEX

```
                               Calendar and AddEvent Macros

                       Ron Goertz
                     NW 223 Clay Ct
                   Pullman  WA  99163
```

These macros are made available for your personal use and no-cost distribution. Permission for any other use must be obtained from the author.

1.2 Introduction to Calendar and AddEvent Macros

INTRODUCTION

A macro for creating calendars in Final Writer v4, and a macro for adding events to those calendars.

The calendar macro by Andrew McCardie is a very versatile macro, allowing calendars in three different formats. The calendars work well for posting on a wall or desk and writing appointments or important dates in. But I wanted more ...

I wanted to print the previous month and next month (mini calendars). I wanted pre-defined holidays (including those on floating days, like Thanksgiving) and special occasions to be inserted automatically when the calendar was created. I wanted to enter single-day events on the calendars with Final Writer. I wanted to enter events stretching across several days or weeks easily. Even though it's my first AREXX program/Final Writer macro, I think Calendar does a fine job of meeting these goals.

Features:
   Mini calendars
   Automatic holidays
   Single-day events
   Multi-day/week events
   Multiple-line events
   Boxed events
   "Shadowed" days from previous/next months

~Calendar~Macro~~~

~User~Variables~

~Highlights~~~~~

~AddEvent~Macro~~~

~User~Variables~

1.3 Installation

INSTALLATION

Copy Calendar.rexx and AddEvent.rexx to your FinalWriter/FWMacros directory (or where ever it is you keep your Final Writer macros). Follow the instructions on page 3-29 of the Final Writer manual to add the macros to the User menu.

1.4 Calendar Macro

CALENDAR MACRO

Select the Calendar macro entry from Final Writer's "User" menu (if you followed the installation procedures included with this guide), otherwise select the "User/Other/ARexx Script ..." menu item and find the Calendar.rexx macro.

The Calendar macro first asks for the desired month (in the format "February"), then the desired year (in the format "1995"). With that info it's off and running.
Highlights
 contained in the macro will be rendered
automatically. If the macro doesn't work for you "out-of-the-box," check the contents of the
User~Variables
 and change them to appropriate values for your
printer setup, font desires, and font paths. Changing the values of the

Program Variables is not recommended.

In order for the AddEvent macro to work, the Month and Year of the calendar is written in the upper left corner (after the December highlight list, see the line: DrawTextBlock 1 100 100 TempDate); as it’s written in white, you won’t see it. THIS MUST BE THE FIRST OBJECT RENDERED TO THE PAGE FOR THE ADDEVENT MACRO TO WORK. If you modify the macro to add additional objects, they MUST be rendered after this TextBlock.

~AddEvent~Macro~

1.5 User Variables

CALENDAR MACRO USER VARIABLES

Calendar user variables ... change according to your printer setup, font preferences, and font paths.

Extended: If set to 0, only days in the current month will be rendered. If set to 1, the last days of the previous month and the first days of the next month will be rendered "shadowed" (i.e., gray print). (Default = 1)

MiniCal: If set to 0, minicalendars will not be rendered. If set to 1, minicalendars will be rendered. (Default = 1)

FullWidth: The width of the paper in inches. (Default = 11.0)

FullHeight: The height of the paper in inches. (Default = 8.5)

TopMargin: The print margin at the top of the page in inches. (Default = 0.5)

BottomMargin: The print margin at the bottom of the page in inches. (Default = 0.2)

LeftMargin: The print margin on the left side of the page in inches. (Default = 0.2)

RightMargin: The print margin on the right side of the page in inches. (Default = 0.2)

ShadowColor: The rendering color of "shadow" text. (Default = "Cyan")

MonthFont: The font for the Month/Year text. (Default = "Serif/Dragon")

MonthFSize: Point size of the Month/Year font. (Default = 44)

MonthFWidth: Percent width of the Month/Year font. (Default = 150)

WeekdayFont: The font for the Weekday text. (Default = "Serif/Dragon")

WeekdayFSize: Point size of the Weekday font. (Default = 22)

WeekdayFWidth: Percent width of the Weekday font. (Default = 80)

DateFont: The font for the Date numbers. (Default = "SoftSans")

DateFSize: Point size of the Date font. (Default = 10)

DateFWidth: Percent width of the Date font. (Default = 100)

HighlightSize: Point size for Highlights (see below); DateFont is used. (Default = 8)

MiniCalSize: Point size of font for mini calendars; DateFont is used. (Default = 8)

DateDown: Distance from top of date box that date number is rendered, in inches. (Default = 0.12)

DateOffset: Distance from left of date box that date number is rendered, in inches. (Default = 0.05)

HighlightOffset: Distance from date number that highlight text is rendered, in inches. (Default = 0.2)

MiniCalWidth: The width of the mini calendars, in inches. (Default = 1.5)

~Highlights~

1.6 Highlights

HIGHLIGHTS

"Highlight" is the term I’m using to describe holidays, anniversaries, birthdays, and other notable dates. Up to 365 highlights can be assigned, one for each day of the year. There is a comment line beginning the highlight list for each month; this is not necessary for the macro but helps organize the highlights. The highlight format is:

```
Highlight.M.D = "Your Text"
```

where M is the month (1 to 12), D is the day (1 to 28/29/30/31), and "Your Text" is what you want to print as the highlight. See the macro for examples of defining the highlights for floating days.

~User~Variables~

1.7 AddEvent Macro

ADDEVENT MACRO

Load a previously created calendar into Final Writer, then select the "AddEvent" item from the "User" menu (if you didn’t follow the installation procedures included with this guide, you may have to select the "User/Other/ARexx Script ..." menu item and find the AddEvent macro).

The AddEvent macro asks you to

```
                Enter event as: Day,Day,Line,Box,Event
```

where

Day: the first day of the event (1 to 28/29/30/31). If preceded by "P", the date will be converted to that date of the PREVIOUS month (e.g., in a calendar for June, "P31" would be converted to May 31.) Note that the Extended variable needn’t be set in the Calendar macro for events to be allowed in the last days of the previous month.

Day: the last day of the event (1 to 28/29/30/31). If the event is single-day, this entry may be omitted but the comma MUST be included. If preceded by "N", the date will be converted to that date of the NEXT month (e.g., in a calendar for June, "N1" would be converted to July 1.) Note that the Extended variable needn’t be set in the Calendar macro for events to be allowed in the first days of the next month.

Line: the row in the date box in which the event is to be placed. Range is from 1 to about 9, depending on font size.

Box: "Y" if the event is to be outlined with a box, anything else (or nothing, but the delimiting comma MUST be included) to leave off the box. Note that a box *will* be added to multi-day/week events and to multi-line events.

Event: the event text, limited to 34 characters by Final Writer. Multiple text lines can be created by inserting ’//’ where the line break is to be.

For example:

```
   12,14,3,,Spring Break//No School!
```

will result in

```
  ---------------------------
  |      Spring Break       |
  |       No School!        |
  ---------------------------
```

being printed across days 12-14 in the current month.

This macro is designed to work specifically with my Calendar macro; it reads the Month/Year from the calendar to determine on which day of the week the month begins. If the macro doesn’t work for you "out-of-the-box," check the contents of the

User~Variables

and change them to appropriate values for your printer setup and font paths/desires. The User Variables in the AddEvent macro must match the Calendar macro's User~Variables to insure the events will be aligned correctly; the Font should be the same as the Calendar macro's DateFont. Changing the values of the Program Variables is NOT recommended.

The AddEvent requestor will come up again after each event is rendered; Select "Cancel" to stop adding events.

Entries must be deleted through Final Writer itself (select the text object and delete it).

~Calendar~Macro~

1.8 User Variables

ADDEVENT MACRO USER VARIABLES

AddEvent user variables must match the Calendar macro's User~Variables to insure the events will be aligned correctly; the Font should be the same as the Calendar macro's DateFont.

```
    FullWidth: The width of the paper in inches. (Default = 11.0)

   FullHeight: The height of the paper in inches. (Default = 8.5)

    TopMargin: The print margin at the top of the page in inches.
               (Default = 0.5)

 BottomMargin: The print margin at the bottom of the page in inches.
               (Default = 0.2)

   LeftMargin: The print margin on the left side of the page in inches.
               (Default = 0.2)

  RightMargin: The print margin on the right side of the page in inches.
               (Default = 0.2)

         Font: The font for the Event. (Default = "SoftSans")

        FSize: Point size of the Event font. (Default = 8)

       FWidth: Percent width of the Event font. (Default = 100)

   TextOffset: Distance from left of date box that Event is rendered, in
               inches. (Default = 0.05, should match DateOffset of
               Calendar macro)
```