WALLCALENDAR USER MANUAL

v1.3 • 17th October 2017 • link

This is the User Manual for the wallcalendar class. Source documentation is in wallcalendar-code.pdf. Clone or download from Github:
https://github.com/profound-labs/wallcalendar/

The document class comes with the following layouts:

- Full page photo, the calendar days overlaid with opacity
- Full page photo, the photo above the calendar days
- Small landscape photo, with a calendar grid

Load event marks from CSV file

Year planner

Thumbnails and captions
CONTENTS

Section 1 · Tutorial: Forest Calendar 3

Section 2 · Tutorial: Translations 8

Section 3 · Tutorial: Load Events from CSV 14

Section 4 · Example: Year Planner Page 16

Section 5 · Example: Photo Thumbnails Page 18

Section 6 · Documentclass Options 22

Section 7 · User Commands 22

Section 8 · Page Layout 24

Section 9 · Contact 24
1 TUTORIAL: FOREST CALENDAR

In this tutorial we will produce the three example pages seen in the summary.

Set the parameters of the month pages in advance, either in the preamble or in the document body, but before calling `\MonthPage{ month }` to typeset it.

A month page can have four areas:
- Photo
- Quote
- Calendar
- Events

Their parameters are set separately for each month:

\begin{verbatim}
\SetPhoto[ options ]{ month }
\SetQuote[ options ]{ month }{ quote text }
\SetCalendar[ options ]{ month }
\SetEvents[ options ]{ month }{ calendar tikz marks }{ events text }
\end{verbatim}

The month page will be typeset with:

\begin{verbatim}
\MonthPage[ options ]{ month }
\end{verbatim}

1.1 DOCUMENTCLASS

To start, load the documentclass and set year, language and the imageFolder:

\begin{verbatim}
\documentclass[
  year=2018,
  language=english,
  imageFolder=./photos/,
]{wallcalendar}
\end{verbatim}

Let's start the preamble with `\makeatletter` to be safe.

\begin{verbatim}
\makeatletter
\end{verbatim}

1.2 FONT SETTINGS

For this example we'll use \TeX\ Gyre Pagella as the main typeface. We also load DejaVu Sans to use a particular glyph as a mark in the calendar (U+263C white sun with rays).

\begin{verbatim}
\usepackage{fontspec}
\defaultfontfeatures{Ligatures={TeX}}
\setmainfont{TeX Gyre Pagella}
\newfontfamily{dejaVuSans}{DejaVu Sans}
\end{verbatim}
I shall set forth for somewhere,
I shall make the reckless choice
Some day when they are in voice
And tossing so as to scare
The white clouds over them on.
I shall have less to say,
But I shall be gone.

\textit{The Sound of the Trees} by Robert Frost
Here we use a conditional to use a different calendar style when `showframe` is turned on, this helps with debugging or tuning the position.

```
\ifshowframe
  \SetCalendar[bg/.style={opacity=0.5, fill=white}]{June}
\else
  \SetCalendar[bg/.style={opacity=0.8}]{June}
\fi
```

Events for particular days are printed under the calendar.

```
\SetEvents{June}{% 
  if (equals=2018-06-21) 
    [day text={\dejaVuSans\char"263C}]; % U+263C white sun with rays
  }{% 
    \raggedleft
    {\dejaVuSans\char"263C} June 21: Summer Solstice
}
```

1.4 July

![Image of a path in a forest]

Same as June, but we will set the image to be placed above the calendar, and we add a transparent background for the quote.

This layout is a good option when the top or the bottom of the photo has to be cropped, and you can’t use the full page aspect ratio for the photo.

```
\SetPhoto[bleed=3mm, file={obscure-crop}]{July}
\SetQuote{% 
  xoffset=0.5linewidth - 0.5\paperwidth -3mm, 
  yoffset=-20mm, 
}{July}{% 
```
I shall have less to say,
But I shall be gone.

*The Sound of the Trees* by Robert Frost

This layout works for photos that are horizontal (landscape orientation), scaled into the bleed margin on three sides.
I shall have less to say,\ 
But I shall be gone.

\textit{The Sound of the Trees} by Robert Frost

\ifshowframe
   \SetCalendar[bg/.style={opacity=0.5}]{August}
\else
   \SetCalendar[bg/.style={opacity=1}]{August}
\fi
\SetEvents{August}{
   if (equals=2018-08-12) [day text={\dejaVuSans\char“263C}];
}{
   \raggedleft
   {\dejaVuSans\char“263C} August 12, 13: Perseids Meteor Shower
}

End of the preamble.

\makeatother

1.6 The document

Typesetting the month pages in the document is now just this much:

\begin{document}
\MonthPage[layout=full page, put photo=full page]{June}
\MonthPage[layout=full page, put photo=full width above calendar]{July}
\MonthPage[layout=small landscape, put photo=full width]{August}
\end{document}
2 TUTORIAL: TRANSLATIONS

In this tutorial we will produce the same calendar in three languages: Japanese, English and Hungarian.

We are going to use IPAPMincho font for the Japanese.

2.1 Files

The main document files:

- cal-translations-japanese.tex
- cal-translations-english.tex
- cal-translations-hungarian.tex

Fonts, formatting settings, etc.:

- local-japanese.sty
- local-english.sty
- local-hungarian.sty

Translation text input:

- frog-japanese.tex
- frog-english.tex
- frog-hungarian.tex

Setup month pages (same across translations):

- frog.tex
2.2 Translations setup

Create the `frog-english.tex` file and use the \SetTxt{ key }{ content } command to set text content for translation keys.

**frog-japanese.tex**

```latex
\SetTxt{September Quote}{% 
\mincho
\Large\textit{araike ya / kawazu tobikomu / oto mo nashi}}
\mincho
\setlength{\parindent}{0pt}
\newcommand\SeptMarks{%
if (equals=2018-09-21)
  \[day text= {\dejaVuSans\char"263C}; % U+263C white sun with rays
}\}
\SetTxt{Sept Events}{%
\dejaVuSans\char"263C} {\mincho 21: 21}
```  

**frog-english.tex**

```latex
\SetTxt{September Quote}{% silent new lake\ 
let the frog jump\ 
not even a sound
\textit{-- Ryokan}}%
\setlength{\parindent}{0pt}
\newcommand\SeptMarks{%
if (equals=2018-09-21)
  \[day text= {\dejaVuSans\char"263C}; % U+263C white sun with rays
}\}
\SetTxt{Sept Events}{% \dejaVuSans\char"263C} Sept 21: September 21
```  

**frog-hungarian.tex**

```latex
\SetTxt{September Quote}{% hallgat az új tó\ 
ugorhat béka belé\ 
vize se csobban
\textit{-- Rjókan}}%
\setlength{\parindent}{0pt}
\newcommand\SeptMarks{%
if (equals=2018-09-21)
  \[day text= {\dejaVuSans\char"263C}; % U+263C white sun with rays
}\}
```
\SetTxt{Sept Events}{\p{dejaVuSans}\char"263C} Szept 21: Szeptember 21
}

NOTE: Using \SetTxt{} to store values intended as tikz marks on the calendar will not work. The \txt{} command will be the value of \@eventmarks and tikz can’t resolve it there.

Put the calendar marks in a command instead, as above with \SeptMarks.

\calendar (cal#1)
  [alntak, dates=\CalendarYear-#1-01 to \CalendarYear-#1-last]
  \@eventmarks;%

% NOTE This code below will not work.
% Put the calendar marks in a command instead.
\SetTxt{Sept Marks}{%
  if (equals=2018-09-21)
    [day text={\p{dejaVuSans}\char"263C}];% U+263C white sun with rays
  }
%
% ...
\SetEvents{September}{%
\txt{Sept Marks}
%}
\raggedleft
\txt{Sept Events}
%

2.3 Document setup

Load the documentclass. We are setting the translations option to define the file where translation keys are set. This file is loaded by the documentclass as an \input.

\documentclass[
  year = 2018,
  language = japanese,  
  translationsInputFile = frog-japanese.tex, 
  imageFolder = ./photos/,
]{wallcalendar}

\usepackage{local-japanese}

% Content is the same across translations
\input{./frog.tex}

\begin{document}

% Just one month
\MonthPage[layout=full page, put photo=full page]{September}
local-japanese.sty

\ProvidesPackage{local-japanese}
\usepackage{fontspec}
\defaultfontfeatures{Ligatures={TeX}}
\setmainfont{TeX Gyre Pagella}
\newfontfamily{dejaVuSans}{DejaVu Sans}
% Japanese font
\newfontfamily{mincho}{IPAPMincho}

% Renew formatting hooks to use the mincho font
\renewcommand\fullPageFmt{%
  \renewcommand*{\monthFmt}{\LARGE\mincho}%
  \renewcommand*{\yearFmt}{\LARGE\mincho}%
  \renewcommand*{\dayLetterColor{}}%
  \renewcommand*{\dayTextFmt}{\small}%
  \renewcommand*{\headingFmt}{%}
  \renewcommand*{\calendarFmt}{%}
  \renewcommand*{\eventsFmt}{%
    \setlength{\parindent}{0pt}\raggedleft\footnotesize%
  }%
}%

local-english.sty

\ProvidesPackage{local-english}
\usepackage{fontspec}
\defaultfontfeatures{Ligatures={TeX}}

cal-translations-english.tex

\documentclass[year = 2018, language = english, translationsInputFile = frog-english.tex, imageFolder = ./photos/]{wallcalendar}
\usepackage{local-english}
\input{./frog.tex}
\begin{document}
\MonthPage[layout=full page, put photo=full page]{September}
\end{document}
cal-translations-hungarian.tex

\documentclass{
  year = 2018,
  language = hungarian,
  translationsInputFile = frog-hungarian.tex,
  imageFolder = ./photos/,
}\{wallcalendar\}
\usepackage{local-hungarian}
\input{./frog.tex}
\begin{document}
\MonthPage[layout=full page, put photo=full page]{September}
\end{document}

local-hungarian.sty

\ProvidesPackage{local-hungarian}
\usepackage{fontspec}
\defaultfontfeatures{Ligatures={TeX}}
\setmainfont{TeX Gyre Pagella}
\newfontfamily{DejaVu Sans}

frog.tex

\makeatletter
\SetPhoto{bleed=3mm, file={frog-crop}}{September}

Use the \texttt{\key{}} command to load text from translation keys:

\SetQuote{xOffset=0pt, yOffset=-140mm}{September}{%
\raggedleft\setlength{\parskip}{10pt}\HUGE\color{white}%
\txt{September Quote}%
}

Calendar settings for the month, using \texttt{\txt} to access translated parts.

\SetCalendar[bg/.style={opacity=0.4}]{September}
\SetEvents{September}{%}
\SeptMarks%
3 TUTORIAL: LOAD EVENTS FROM CSV

3.1 CSV FILES

Events in the CSV should be already sorted by date.

If you are using more than one CSV, put all events with notes (i.e. indexed entries) in the same CSV. The index number of the mark is taken from the row number in the CSV, so a second CSV with notes would start the count from 1 again.

We're going to use the following csv files, see in the ./doc/examples/data/ folder.

- holidays.csv
- moonphases.csv
- mark_defaults.csv

3.2 EVENT FORMATTING

You can format the event output by setting the format cmd key:

```plaintext
\parseMonthEvents[format cmd = {
  \textsuperscript{\eMark}~\eMonthShort~\eDay: & \eNote\ifnumless\eIdx\eMaxIdx{, .}\\
},]
```

Or define a Lua formatting function and set it with the format func key:

```plaintext
helpers.lua

function eventFmtCustom(idx, max_idx, event, event_date, mark)
  local d = event_date
tex.sprint(string.format(
    "%\textsuperscript{\%s} & \%s %s: & %s \\",
    mark.symbol, d:fmt("%b"), d:getday(), event.note
  ));
end

\parseMonthEvents[format func = eventFmtCustom]
```

3.3 Document setup

cal-marks.tex

\documentclass[
    year = 2018,
    eventsCsv = ./data/holidays.csv,
    markDefaultsCsv = ./data/mark_defaults.csv,
    imageFolder = ./photos/,
]{wallcalendar}

\makeatletter
\colorlet{mooncolor}{darkgold}
\usepackage{fontspec}
\defaultfontfeatures{Ligatures={TeX}}
\setmainfont{TeX Gyre Pagella}
\newfontfamily{dejaVuSans}{DejaVu Sans}
\SetPhoto[bleed=3mm, file={frog-crop}]{September}
\SetQuote[xOffset=0pt, yOffset=-140mm]{September}{
\raggedleft\setlength{\parskip}{10pt}\HUGE\color{white}\silent new lake\\
let the frog jump\\
not even a sound
}\textit{-- Ryokan}\%
}
\SetCalendar[bg/.style={opacity=0.4}]{September}
\SetEvents{September}{
\parseMonthMarksDayTextUsing{./data/moonphases.csv}\%
\parseMonthMarksDayText\%
\parseMonthMarksNote\%
}\%
\raggedright
\parseMonthEvents[%
\textsuperscript{\eMark}\eMonthShort-\eDay:\space%
\eNote\ifnumless{\eIdx}{\eMaxIdx}{,\space}\%}
]
\makeatother
\begin{document}
\MonthPage[layout=full page, put photo=full page]{September}
\end{document}
4 EXAMPLE: YEAR PLANNER PAGE

4.1 Document setup

cal-year-planner.tex

\documentclass[
  year = 2018,
  eventsCsv = ./data/holidays.csv,
  markDefaultsCsv = ./data/mark_defaults.csv,
  imageFolder = ./photos/]
{wallcalendar}
\makeatletter
\usepackage{fontspec}
\defaultfontfeatures{Ligatures={TeX}}
\setmainfont{TeX Gyre Pagella}
\newfontfamily{dejaVuSans}{DejaVu Sans}
\renewcommand\\wall@plm[1]{% \luadirect{
  require("wallcalendar-helpers.lua")
  monthMarksDayText(\luastring(#1), nil, \luastring{\plannerMarksDayTextCSV})
  monthMarksDayText(\luastring(#1), nil, \luastring{./data/moonphases.csv})
  tex.sprint(\text{'};\text{'}
})
}

4.2 \YearPlannerPage

\newcommand{\plannerYearFmt}{\fontsize{26}{26}\selectfont\color{orangegold}}
\newlength{\plannerNotesSep}
\setlength{\plannerNotesSep}{3mm}
\newcommand{\preYearPlannerPageHook}{%
  \setlength{\markNumberAbove}{-9pt}}%
4.3 **Use it**

\begin{document}
\YearPlannerPage
\end{document}
5 EXAMPLE: PHOTO THUMBNAI LS PAGE

5.1 Document setup

cal-thumbnails.tex

\documentclass[year=2018, imageFolder=./photos/]{wallcalendar}
\makeatletter
\usepackage{fontspec}
\defaultfontfeatures{Ligatures={TeX}}
\setmainfont{TeX Gyre Pagella}
\newfontfamily{dejaVuSans}{DejaVu Sans}
\newlength{wall@tmp@a}
\newlength{wall@tmp@b}
\setlength{\@wall@thumbWidth}{0.1749\calPaperWidth}% 30mm at the 6.75in page width, 0.1749 = 1/5.715
\setlength{\@wall@thumbHeight}{\@wall@thumbWidth}
\setlength{\@wall@thumbCaptionWidth}{0.2333\calPaperWidth}% 40mm at 6.75in page width
\newcommand{\thumbFmt{}}

5.2 \ThumbWithCaptionLeftSide

Typesets the photo thumb image with its caption text on the left side.

\ThumbWithCaptionLeftSide(January)
\newcommand{\ThumbWithCaptionLeftSide}[1]{
% \pgfkeys{/Photo/#1/thumbFile/.get=\@wall@thumbFile}
% \ifx\@wall@thumbFile\empty
%   \pgfkeys{/Photo/#1/file/.get=\@wall@thumbFile}
% \fi
% Thumbnail caption
% \hspace*{3mm}
\begin{minipage}[b][\@wall@thumbHeight][t]{$\@wall@thumbCaptionWidth$}
\raggedleft
\thumbFmt
{\thumbMonthFmt \@tr@monthNumName{\monthToNum(#1)}}
\par
\thumbCaptionFmt{\@wall@photoCaption}
\end{minipage}
\hspace*{3mm}
% Thumbnail photo
\begin{minipage}[b][\@wall@thumbHeight][\@wall@thumbWidth]
% FIXME placeholder
\includegraphics[ keepaspectratio, height=\@wall@thumbHeight ]{\@wall@thumbFile}
\end{minipage}
}

5.3 \ThumbWithCaptionRightSide

Typesets the photo thumb image with its caption text on the right side.

\ThumbWithCaptionRightSide(January)
5.4 \ThumbsPage

\newlength{\thumbColumnWidth}
\newlength{\thumbColumnHeight}
\newlength{\thumbSep}
\newlength{\@t@a}
\newlength{\@t@b}

% vertical spacing
\setlength{\@t@a}{0.2\textheight}

% horizontal spacing
\setlength{\@t@b}{0.2\@wall@thumbCaptionWidth}\% 8mm at 40mm caption width
\ifdimcomp{\@t@a}{<}{\@t@b}{
  \setlength{\thumbSep}{\@t@a}
}\{  
  \setlength{\thumbSep}{\@t@b}
}\}
\setlength{\thumbColumnWidth}{0.5\calPaperWidth - \thumbSep}
\setlength{\thumbColumnHeight}{\textheight}
\newcommand\ThumbsPage{\%
\clearpage
\hspace*{-\@wall@leftMargin}\%
\hspace*{-1pt}\% small correction, space gets in somewhere
\% Wrap
\begin{minipage}[t]{\thumbColumnHeight}{\calPaperWidth}\%
\centering
\setlength{\parindent}{0pt}\%
\setlength{\parskip}{0pt}\%
\% NOTE: multicols will not \vfill
\% Left Column
\begin{minipage}[t]{\thumbColumnHeight}{\thumbColumnWidth}\%
\raggedleft
\ThumbWithCaptionLeftSide{January}\%
\vspace*{\thumbSep}\%
\ThumbWithCaptionLeftSide{March}\%
\vspace*{\thumbSep}\%
\ThumbWithCaptionLeftSide{May}\%
\vspace*{\thumbSep}
5.5 Setup the photo keys

\SetPhoto[file={frog-crop}, thumbFile={frog-crop-thumb}, caption={Let the frog jump}]{January}
\SetPhoto[file={frog-crop}, thumbFile={frog-crop-thumb}, caption={Let the frog jump}]{February}
\SetPhoto[file={frog-crop}, thumbFile={frog-crop-thumb}, caption={Let the frog jump}]{March}
\SetPhoto[file={frog-crop}, thumbFile={frog-crop-thumb}, caption={Let the frog jump}]{April}
\SetPhoto[file={frog-crop}, thumbFile={frog-crop-thumb}, caption={Let the frog jump}]{May}
\SetPhoto[file={frog-crop}, thumbFile={frog-crop-thumb}, caption={Let the frog jump}]{June}
\SetPhoto[file={frog-crop}, thumbFile={frog-crop-thumb}, caption={Let the frog jump}]{July}
\SetPhoto[file={frog-crop}, thumbFile={frog-crop-thumb}, caption={Let the frog jump}]{August}
5.6 Use it

\begin{document}
\ThumbsPage
\end{document}

6 DOCUMENTCLASS OPTIONS

year

language

imageFolder

showframe

showtrims

varnishmask

7 USER COMMANDS

\SetPhoto[ options ]{ month }
8 PAGE LAYOUT

We will review how the page layout happens when we call \MonthPage.

Before calling \MonthPage to typeset a particular month, be sure that the content for the given month has been already set with \SetPhoto, \SetQuote, \SetCalendar and \SetEvents.

Then we can typeset that month, for example:

\MonthPage[layout=full page, put photo=full page]{June}

This will be a full page portrait month with the photo scaled to cover the entire page with additional bleed on all four sides.

The above showframe figure shows the structural elements of the page.

Every layout is implemented by a single handler macro which will deal with all the typesetting of the given page. The full page key is set to the \@wall@fullPageLayout macro by default, and so this gets called.

The layout macro is just a free-style placeholder. It can access the photo, quote, calendar and events as set earlier, but it is up to the macro to implement what to do with them.

This is for the convenience of setting the page elements using the same interface, but being able to execute different layouts for different pages.

The class contains two layout examples. The full page layout is best for portrait photos that can be scaled to cover the entire page. The small landscape layout is for landscape photos which can be scaled horizontally, possibly bleeding into the side margins.

9 CONTACT

Github: https://github.com/profound-labs/wallcalendar/

Email: Gambhīro Bhikkhu <gambhiro.bhikkhu.85@gmail.com>