

WALLCALENDAR USER MANUAL

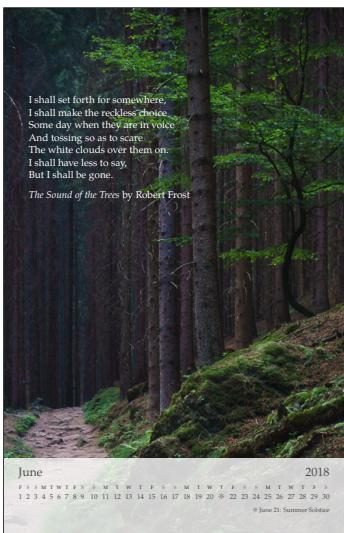
v1.4.0 · 10th November 2018 · [link](#)

This is the **User Manual** for the `wallcalendar` class. **Source documentation** is in `wallcalendar-code.pdf`. Clone or download from Github:

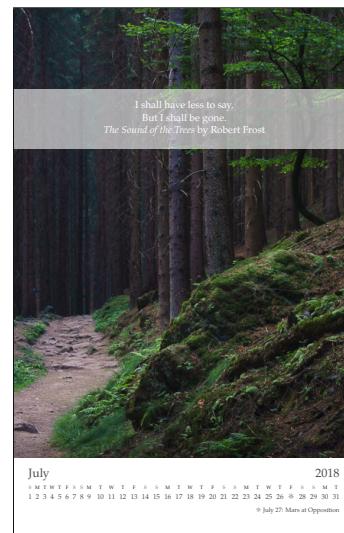
<https://github.com/profound-labs/wallcalendar/>

The documentclass comes with the following layouts:

Full page photo, the calendar days overlaid with opacity



Full page photo, the photo above the calendar days



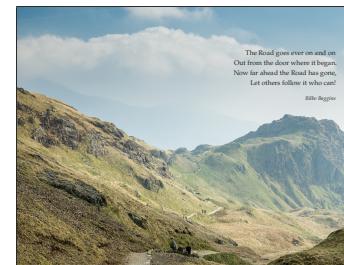
Small landscape photo, with a calendar grid



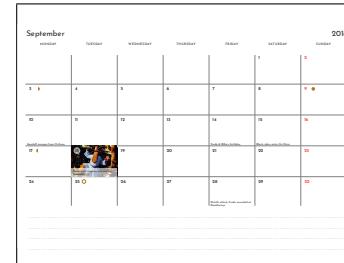
Title page



Photo and notes on two pages



Year planner



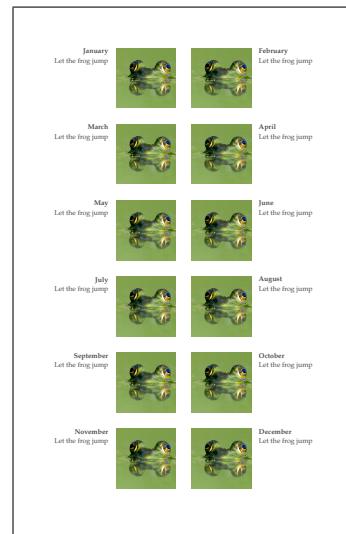
2018											
January					February					March	
M	T	W	T	F	S	S	M	T	W	T	F
1	○	3	4	5	6	7	1	2	3	4	5
2	○	9	10	11	12	13	14	5	6	7	8
3	○	15	16	17	18	19	20	12	13	14	15
4	○	21	22	23	24	25	26	19	20	21	22
5	○	28	29	30	31		27	28	29	30	31
6	○						26	27	28	29	30
April											
M	T	W	T	F	S	S	M	T	W	T	F
1	○	1	2	3	4	5	6	7	8	9	10
2	○	14	15	16	17	18	19	20	21	22	23
3	○	21	22	23	24	25	26	27	28	29	30
4	○	28	29	30	31		29	30	31	1	2
5	○						28	29	30	31	1
May											
M	T	W	T	F	S	S	M	T	W	T	F
1	○	1	2	3	4	5	6	7	8	9	10
2	○	14	15	16	17	18	19	20	21	22	23
3	○	21	22	23	24	25	26	27	28	29	30
4	○	28	29	30	31		28	29	30	31	1
5	○						27	28	29	30	31
June											
M	T	W	T	F	S	S	M	T	W	T	F
1	○	1	2	3	4	5	6	7	8	9	10
2	○	14	15	16	17	18	19	20	21	22	23
3	○	21	22	23	24	25	26	27	28	29	30
4	○	28	29	30	31		27	28	29	30	31
5	○						26	27	28	29	30
July											
M	T	W	T	F	S	S	M	T	W	T	F
1	○	1	2	3	4	5	6	7	8	9	10
2	○	14	15	16	17	18	19	20	21	22	23
3	○	21	22	23	24	25	26	27	28	29	30
4	○	28	29	30	31		27	28	29	30	31
5	○						26	27	28	29	30
August											
M	T	W	T	F	S	S	M	T	W	T	F
1	○	1	2	3	4	5	6	7	8	9	10
2	○	14	15	16	17	18	19	20	21	22	23
3	○	21	22	23	24	25	26	27	28	29	30
4	○	28	29	30	31		27	28	29	30	31
5	○						26	27	28	29	30
September											
M	T	W	T	F	S	S	M	T	W	T	F
1	○	1	2	3	4	5	6	7	8	9	10
2	○	14	15	16	17	18	19	20	21	22	23
3	○	21	22	23	24	25	26	27	28	29	30
4	○	28	29	30	31		27	28	29	30	31
5	○						26	27	28	29	30
October											
M	T	W	T	F	S	S	M	T	W	T	F
1	○	1	2	3	4	5	6	7	8	9	10
2	○	14	15	16	17	18	19	20	21	22	23
3	○	21	22	23	24	25	26	27	28	29	30
4	○	28	29	30	31		27	28	29	30	31
5	○						26	27	28	29	30
November											
M	T	W	T	F	S	S	M	T	W	T	F
1	○	1	2	3	4	5	6	7	8	9	10
2	○	14	15	16	17	18	19	20	21	22	23
3	○	21	22	23	24	25	26	27	28	29	30
4	○	28	29	30	31		27	28	29	30	31
5	○						26	27	28	29	30
December											
M	T	W	T	F	S	S	M	T	W	T	F
1	○	1	2	3	4	5	6	7	8	9	10
2	○	14	15	16	17	18	19	20	21	22	23
3	○	21	22	23	24	25	26	27	28	29	30
4	○	28	29	30	31		27	28	29	30	31
5	○						26	27	28	29	30

¹ Sep 10: Gandalf escapes from Orthanc. ² Sep 14: Frodo & Bilbo's birthday. ³ Sep 15: Black Riders attack the Shire. ⁴ Sep 16: Frodo and Sam are rescued by the Ents. ⁵ Sep 17: Fall of Minas Tirith. ⁶ Sep 18: Sam and Frodo cross the bridge of Mithril. ⁷ Oct 16: Boromir reaches Rivendell. ⁸ Oct 17: Council of Elrond.

Load event marks from CSV file



Thumbnails and captions



CONTENTS

Section 1 · Tutorial: Forest Calendar	4
Section 2 · Tutorial: Translations	9
Section 3 · Tutorial: Load Events from CSV	15
Section 4 · Example: Photo and Notes with Title page	17
Section 5 · Example: Year Planner Page	20
Section 6 · Example: Photo Thumbnails Page	22
Section 7 · Documentclass Options	26
Section 8 · User Commands	26
Section 9 · Page Layout	28
Section 10 · Contact	28

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:

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

The month page will be typeset with:

```
\MonthPage[ options ]{ month }
```

1.1 DOCUMENTCLASS

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

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

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

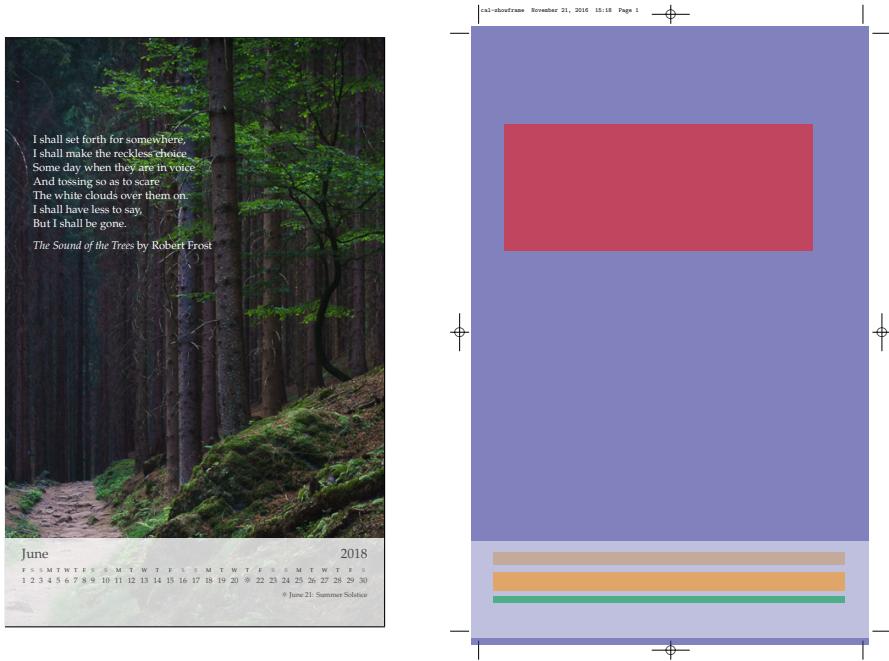
```
\makeatletter
```

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).

```
\usepackage{fontspec}
\defaultfontfeatures{Ligatures={TeX}}
\setmainfont{TeX Gyre Pagella}
\newfontfamily\dejaVuSans{DejaVu Sans}
```

1.3 JUNE



`showtrims` and `showframe` class options
show the page structure.

It will be a full page photo, with 3mm bleed on all four sides. You can see the bleed if you enable the `showtrims` class option. We also specify the file name of the photo (no extension), this will be the argument of `\includegraphics`.

```
\SetPhoto[bleed=3mm, file={obscure-crop}]{June}
```

A quote will be positioned over the photo. The quote is in a `\linewidth` wide minipage, attached to the top left corner of the page. Use `\raggedleft`, `\raggedright`, or `\centering` for alignment, and the `xOffset` and `yOffset` options to move the quote's minipage to the exact position.

```
\SetQuote[xOffset=-5mm, yOffset=-20mm]{June}{%
\raggedright
\setlength{\parskip}{10pt}%
\Large
\color{white}

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}
}
```

The layout macro will place the calendar at the bottom, dates in a single line.

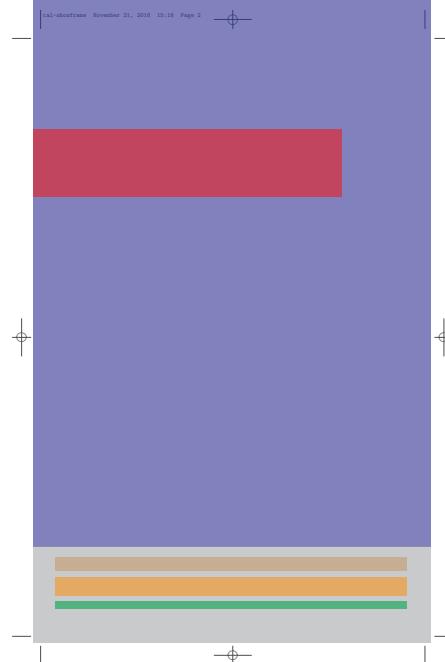
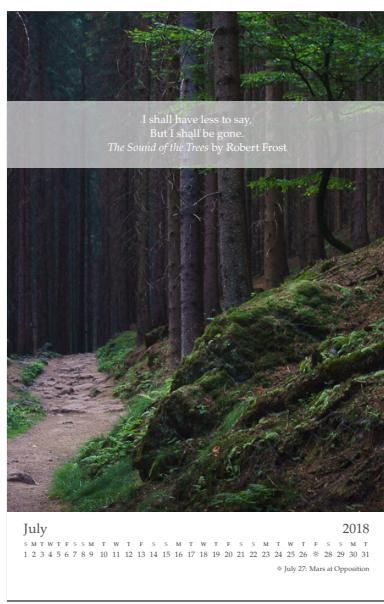
Here we use a conditional to use a different calendar style when `showframe` is turned on, this helps with debugging or tuning the position.

```
\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



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[%]
  x0Offset=0.5\linewidth - 0.5\paperwidth -3mm,
  y0Offset=-20mm,
]{}{July}{%
```

```
\begin{tikzpicture}%
\node [
    fill=white, opacity=0.6, minimum width=\paperwidth + 3mm,
    minimum height=30mm] {};
\node [] {%
\begin{minipage}{\paperwidth + 3mm}%
\centering
\Large
\color{white}
I shall have less to say, \\
But I shall be gone.

\textit{The Sound of the Trees} by Robert Frost
\end{minipage}%
};

\end{tikzpicture}%

\ifshowframe
\SetCalendar[bg/.style={opacity=0.5}]{July}
\else
\SetCalendar[bg/.style={opacity=1}]{July}
\fi

\SetEvents{July}%
\if \equals=2018-07-27 [day text={\dejaVuSans\char"263C}];%
\raggedleft
{\dejaVuSans\char"263C} July 27: Mars at Opposition
}
```

1.5 AUGUST



This layout works for photos that are horizontal (landscape orientation), scaled into the bleed margin on three sides.

```
\SetPhoto[bleed=3mm, file={obscure-crop}, yOffset=-150mm]{August}

\SetQuote[yOffset=-3mm]{August}{%
\centering
\setlength{\parskip}{10pt}%
\Large
\color{black!80}

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`

```
\SetTxt{September Quote}{%
{\mincho }

{\Large\textit{araike ya / kawazu tobikomu / oto mo nashi}}

{\mincho }%
}

\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`

```
\SetTxt{September Quote}{%
silent new lake\\
let the frog jump\\
not even a sound

\textit{-- Ryokan}%

\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`

```
\SetTxt{September Quote}{%
hallgat az új tó\\
ugorhat béka belé\\
vize se csobban

\textit{-- Rjókan}%

\newcommand\SeptMarks{%
if (equals=2018-09-21)
[day text={\dejaVuSans\char"263C}];% U+263C white sun with rays
}
```

```
\SetTxt{Sept Events}{%
{\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)
[alnitak, 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={\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`.

`cal-translations-japanese.tex`

```
\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}
```

```
\end{document}
```

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*\dayLetterFmt{\tiny\mincho}%
  \renewcommand*\dayTextFmt{\small}%
  \renewcommand*\quoteFmt{}%
  \renewcommand*\headingFmt{\centering}%
  \renewcommand*\calendarFmt{\centering}%
  \renewcommand*\eventsFmt{%
    \setlength{\parindent}{0pt}\raggedleft\footnotesize%
  }%
}
```

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}
```

local-english.sty

```
\ProvidesPackage{local-english}

\usepackage{fontspec}
\defaultfontfeatures{Ligatures={TeX}}
```

```
\setmainfont{TeX Gyre Pagella}
\newfontfamily\dejaVuSans{DejaVu Sans}
```

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\dejaVuSans{DejaVu Sans}
```

frog.tex

```
\makeatletter

\SetPhoto[bleed=3mm, file={frog-crop}]{September}
```

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

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

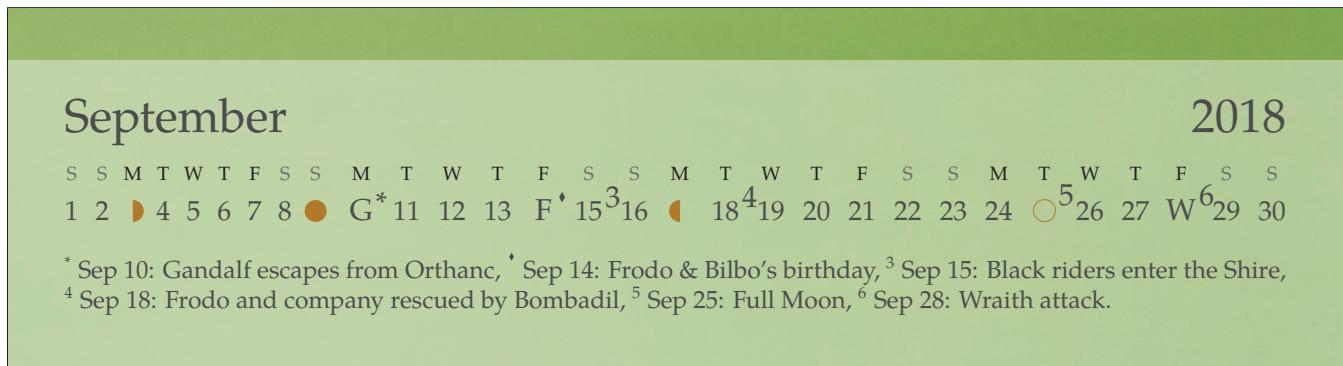
Calendar settings for the month, using \txt to access translated parts.

```
\SetCalendar[bg/.style={opacity=0.4}]{September}

\SetEvents{September}{%
\SeptMarks%
```

```
{%  
\raggedleft  
\txt{Sept Events}  
}  
  
\makeatother
```

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:

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

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

`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{%
    format cmd = {%
        \textsuperscript{\eMark-\eMonthShort-\eDay:\space}%
        \eNote\ifnumless{\eIdx}{\eMaxIdx}{,\space}{.}}%
},
}%
}

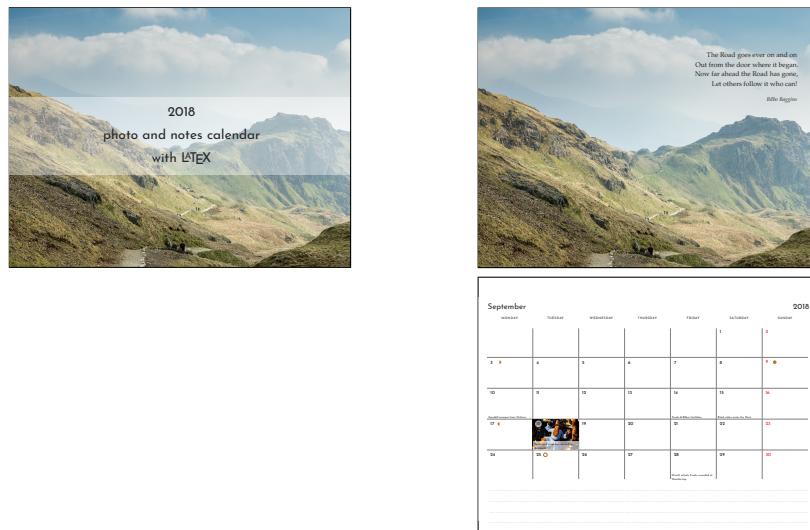
\makeatother

\begin{document}

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

\end{document}
```

4 EXAMPLE: PHOTO AND NOTES WITH TITLE PAGE



See doc/examples/cal-photo-and-notes.tex.

4.1 TITLE PAGE

A title page with a full page photo can be produced with:

```
\TitlePage{photo_path}{overlay_height}{title_text}
```

Crop the photo to the aspect ratio of the trimmed paper size.

Open the image in GIMP, and set “Fixed aspect ratio” on the Rectangle Select tool. It takes integers, so you can use the page dimensions multiplied up. For US Letter (11 x 8.5 inch), this would be 110:85. Then select the area and crop.

It also helps to put guides on the edges by selecting all (Ctrl + A) and using Image > Guides > New Guides from Selection.

4.2 ADDING BACKGROUND PHOTOS TO THE CALENDAR DAYS

This can be done by positioning a \node over the day.

The background photo has to be cropped to the aspect ratio of a grid cell in the calendar.

You can determine the aspect by first creating a calendar PDF with the intended page size, then open the PDF in GIMP. Import the calendar page at 300dpi, and use the selection tool to draw a rectangle over a grid cell.

The pixel width and height of this cell can be used directly to crop the photo with the Rectangle Select tool. Use the cell width and height to set a fixed aspect ratio of the selection area (the aspect matters, not the specific pixel dimensions).

The spacing has to be adjusted for precise positioning, see below.

```
\SetEvents{September}{%
% All this will be in tikzpicture environment, after drawing the calendar.
%
% This adds a semicolon to end the \calendar command so we can start adding nodes.
\semicolonEndTikzCalendar%
%
% Overlaying a photo with text on a specific date in this month
```

```
%  

% photo node  

\node (day-photo) [  

    above right=3.2pt and 0.1pt of cal09-2018-09-18.north west,  

    inner sep=0pt,  

    anchor=north west,  

] {};  

\includegraphics[width={\@t@calendar@dayXshift - 0.25pt}]{TomBombadil-crop};%  

%  

% the photo is dark, add a white background for the day number  

\node (top-whitecircle) [  

    below right=0.3pt and 0.1pt of cal09-2018-09-18-daynumber.center,  

    anchor=center,  

    circle,  

    minimum width=15pt,  

    inner sep=0pt,  

    fill=white,  

    opacity=0.6,  

] {};%  

%  

% the photo is dark, add a white background for the day's event text  

\node (bottom-whitebox) [  

    above left=0pt and 0pt of day-photo.south west,  

    anchor=south west,  

    minimum width={\@t@calendar@dayXshift - 0.25pt},  

    minimum height=18pt,  

    inner sep=0pt,  

    fill=white,  

    opacity=0.6,  

] {};%  

%  

% repeat the day number (the photo has covered it)  

\calDayNumberNode{cal09-2018-09-18}{-redo}{\dayTextFmt 18};%  

%  

% print the day's marks (moon phases in this case) next to the day number  

\parseMonthMarksDayTextInlineUsing{./data/moonphases-2018.csv};%  

%  

% print the day's event text  

\parseMonthInlineNotes%;  

}{};
```

4.3 DISABLING THE NOTEPAD AREA

Formatting commands and sizes have to renewed in \photoAndNotesFmt (add it to the preamble), because every month page first runs a command which resets them to default values, and then runs the above formatting command to change the defaults.

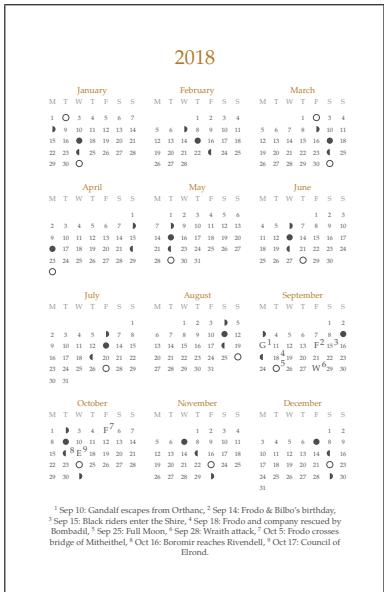
Disable the notepad area by renewing \notepadContent to empty.

In this case the calendar can be taller. Find a month with six rows to experiment by changing \@t@calendar@minimumHeight and \@t@calendar@verticalSpacing.

```
\renewcommand\photoAndNotesFmt{%
  \renewcommand\notepadContent{}%
  \setlength{\@t@calendar@minimumHeight}{\calPaperHeight - 20mm}%
  \setlength{\@t@calendar@verticalSpacing}{30mm}%
}
```

} |

5 EXAMPLE: YEAR PLANNER PAGE



5.1 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}

% Use two CSV files for day text input to include the moon phases
\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(';' )
}}}
```

5.2 \YEARPLANNERPAGE

```
\newcommand{\plannerYearFmt}{\fontsize{26}{26}\selectfont\color{orangelgold} }

\newlength{\plannerNotesSep}
\setlength{\plannerNotesSep}{3mm}

\newcommand{\preYearPlannerPageHook}{%
\setlength{\markNumberAbove}{-9pt} %
```

```

\setlength{\markNumberRight}{-6pt}%
\setlength{\markDayTextAbove}{-11pt}%
\setlength{\markDayTextRight}{-6pt}%
}

\newcommand\postYearPlannerPageHook{%
\setlength{\markNumberAbove}{-10pt}%
\setlength{\markNumberRight}{-3pt}%
\setlength{\markDayTextAbove}{-10pt}%
\setlength{\markDayTextRight}{-3pt}%
}

\newcommand\printPlannerTitle{\plannerYearFmt \CalendarYear}

\newcommand\YearPlannerPage{%
\newpage
\ifvarnishmask
\mbox{}
\else
\preYearPlannerPageHook
{\centering

{\printPlannerTitle}

\vspace*{7mm}

\YearPlannerPortrait

\vspace*{\plannerNotesSep}

\plannerEvents

}
\postYearPlannerPageHook

\fi
}

\makeatother

```

5.3 USE IT

```

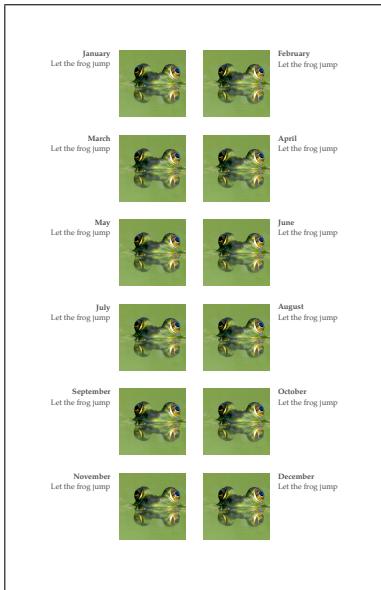
\begin{document}

\YearPlannerPage

\end{document}

```

6 EXAMPLE: PHOTO THUMBNAILS PAGE



6.1 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
```

6.2 \THUMBWITHCAPTIONLEFTSIDE

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

```
\ThumbWithCaptionLeftSide{January}
```

```
\newlength\@wall@thumbWidth
\newlength\@wall@thumbHeight
\newlength\@wall@thumbCaptionWidth
\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{}
```

```
\newcommand\thumbMonthFmt{\fontsize{10}{13}\selectfont\bfseries}
\newcommand\thumbCaptionFmt{\fontsize{10}{13}\selectfont}

\def\@wall@thumbFile{}
\def\@wall@photoCaption{}

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

6.3 \ThumbWithCaptionRightSide

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

```
\ThumbWithCaptionRightSide[January]
```

```
\newcommand\ThumbWithCaptionRightSide[1]{%
\pgfkeys{/Photo/#1/thumbFile/.get=\@wall@thumbFile}%
\ifx\@wall@thumbFile\empty
  \pgfkeys{/Photo/#1/file/.get=\@wall@thumbFile}%
\fi
\pgfkeys{/Photo/#1/caption/.get=\@wall@photoCaption}%
% Thumbnail photo
\begin{minipage}[b][\@wall@thumbHeight]{\@wall@thumbWidth}%
% FIXME placeholder
%\placeholder{%
\includegraphics[ keepaspectratio, height=\@wall@thumbHeight ]{\@wall@thumbFile}%
%}%
\end{minipage}%
\hspace*{3mm}%
% Thumbnail caption
\ifvarnishmask%
\hspace*{\@wall@thumbWidth}%
\else%
```

```
\begin{minipage}[b][\@wall@thumbHeight][t]{\@wall@thumbCaptionWidth}%
\raggedright
\thumbFmt
{\thumbMonthFmt \@tr@monthNumName{\monthToNum{#1}}}\par
{\thumbCaptionFmt \@wall@photoCaption}%
\end{minipage}%
\fi%
}
```

6.4 \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}
\else
  \setlength{\thumbSep}{\@t@b}
\fi

\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}[\thumbColumnHeight]{\thumbColumnWidth}%
\raggedleft

\ThumbWithCaptionLeftSide{January}

\vspace*{\thumbSep}

\ThumbWithCaptionLeftSide{March}

\vspace*{\thumbSep}

\ThumbWithCaptionLeftSide{May}

\vspace*{\thumbSep}

```

```
\ThumbWithCaptionLeftSide{July}

\vspace*{\thumbSep}

\ThumbWithCaptionLeftSide{September}

\vspace*{\thumbSep}

\ThumbWithCaptionLeftSide{November}

% End of left column
\end{minipage}%
%
\hspace*{\thumbSep}%
%
% Right column
\begin{minipage}[t][\thumbColumnHeight]{\thumbColumnWidth}%
\raggedright

\ThumbWithCaptionRightSide{February}

\vspace*{\thumbSep}

\ThumbWithCaptionRightSide{April}

\vspace*{\thumbSep}

\ThumbWithCaptionRightSide{June}

\vspace*{\thumbSep}

\ThumbWithCaptionRightSide{August}

\vspace*{\thumbSep}

\ThumbWithCaptionRightSide{October}

\vspace*{\thumbSep}

\ThumbWithCaptionRightSide{December}

% End of right column
\end{minipage}%

% End of wrap
\end{minipage}%
}
```

6.5 SETUP THE PHOTO KEYS

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

```
\SetPhoto[file={frog-crop}, thumbFile={frog-crop-thumb}, caption={Let the frog jump}] {September}
\SetPhoto[file={frog-crop}, thumbFile={frog-crop-thumb}, caption={Let the frog jump}] {October}
\SetPhoto[file={frog-crop}, thumbFile={frog-crop-thumb}, caption={Let the frog jump}] {November}
\SetPhoto[file={frog-crop}, thumbFile={frog-crop-thumb}, caption={Let the frog jump}] {December}

\makeatother
```

6.6 USE IT

```
\begin{document}

\ThumbsPage

\end{document}
```

7 DOCUMENTCLASS OPTIONS

```
year
```

```
language
```

```
imageFolder
```

```
showframe
```

```
showtrims
```

```
varnishmask
```

8 USER COMMANDS

```
\SetPhoto[ options ]{ month }
```

```
\SetQuote[ options ]{ month }{ quote text }
```

```
\SetCalendar[ options ]{ month }
```

```
\SetEvents[ options ]{ month }{ calendar tikz marks }{ events text }
```

```
\MonthPage[ options ]{ month }
```

9 PAGE LAYOUT

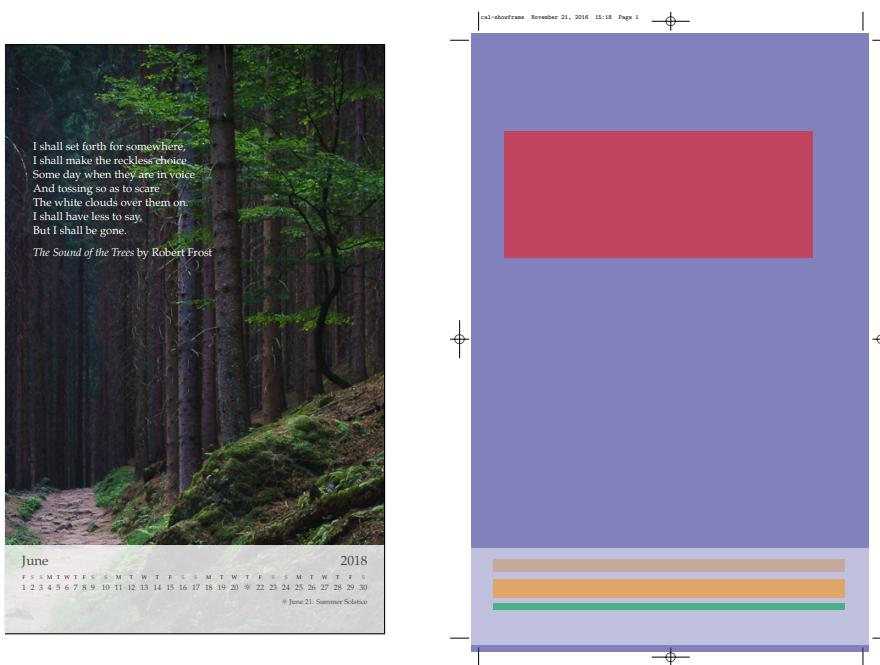
We will review how the page layout happens when we call `\MonthPage`.

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.

10 CONTACT

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

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