Calendars Using the PocketDiary Module

(Examples)

Willi Egger

Now that we have an entire machinery which can calculate calendars for creating PocketDiaries, we can also think about using it for other types of calendars. Here I present a photo calendar, a menu calendar, and a year calendar with sun and moon data as well as two types of year calendars with either a single page per day and the other with 2 pages for one week. The latter four examples are print-impositioned for books.

1. Introduction

Looking at the possibilities of creating calendars for PocketDiary shows that it should be possible to generate calendars of all kinds including outside the PocketDiary module. Loading the module provides the machinery for calculations and the basic formats to create calendar tables.

After having understood how the module works it is a matter of imagination and trials to create new forms of calendars.

2. Photo-Calendar

This projects presents a flip-over type of photo calendar. The top part of each sheet of paper carries a photo, month name and year which appear centered.

Below you get a month-table, where the first row contains weekday names. Sunday is written in red as are the respective day numbers.

For such a calendar you need 13 photos.

First we load the pocketdiary module:

```
\usemodule[pocketdiary]
```

Next if we want a font other than Pagella (the default font for PocketDiary), we must setup the body font:

```
\setupbodyfont[dejavu,ss,12pt]
```

We do not want page numbers:

```
\setuppagenumbering[location=]
```

In this case we prepare an A4 portrait-sized calendar:

```
\setuppapersize[A4,portrait]%[A4,portrait]
```

We adapt the layout of the page to the intended size of the calendar and set the main language:

```
\setuplayout
 [topspace=20mm,
  backspace=8.6mm,
  header=0pt,
  footer=0pt,
  height=middle,
  width=middle]

\mainlanguage[en]
```

The photos are collected in a subdirectory, so we tell ConTEXt where to find them:

```
\setupexternalfigures
  [directory=./calendar-photos]
```

Since the photos are placed with a loop, we assign the file names to symbolic names, numbers in this case:

```
\useexternalfigure
  [1]
  [Birkenteller]

  ...

\useexternalfigure
  [13]
  [Kantele-Rafal]
```

From all the variables used for the PocketDiary we only need to set the year:

```
\setvariables
  [PocketDiary]
  [Year=2023]
```

This example is able to deal with different languages. The label texts are setup as follows:

```
\setuplabeltext[en][fotocal={Photo Calendar}]
\setuplabeltext[nl][fotocal=Fotokalender]
\setuplabeltext[de][fotocal=Fotokalender]
\setuplabeltext[fr][fotocal={Calendrier photo}]
\setuplabeltext[it][fotocal={Calendario fotografico}]
\setuplabeltext[es][fotocal={Calendario fotográfico}]
```

Now we are ready to create the calendar with the help of some Lua code:

```
\startluacode
  local report = logs.reporter("Photo calendar")

  function thirddata.calendar.Photo_calendar_page(y)

    report("Working in function: calendar.Photo_calendar_page")

    local year = y
    for i = 1,12 do

      local monthname = string.lower(os.date("%B",
        os.time{year=year,month=i,day=1}))

      context.startplacefigure({number="",title=""})
         context.externalfigure({i}, {width = "0.8\\textwidth"})
      context.stopplacefigure()
      context.strut()
      context("\\vfill")
      context.startalign({"middle"})
        context("\\bfa")
        context.labeltext(monthname)
        context("~")
        context(year)
      context.stopalign()
      context.blank()

      thirddata.diary.monthtableH(i,year)

      context.page()
    end
  end
\stopluacode
```

The table layout needs a little tweaking, so this layout is wrapped in `\startsetups ... \stopsetups`:

```
\startsetups table:month
   \setupTABLE
     [c]
     [each]
     [width=\dimexpr\textwidth/7\relax,frame=on,style=\tfa]
   \setupTABLE[r][each][height=1.8\lineheight,align={center,lohi}]
   \setupTABLE[c][7][framecolor=black,foregroundcolor=red]
\stopsetups
```

Now we need yet another macro for the creation of the calendar:

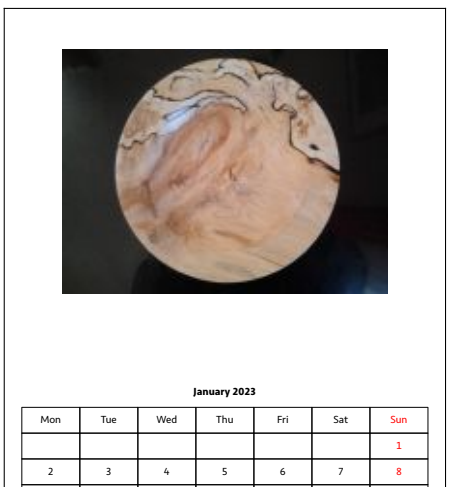
```
\define[1]\Photocalendarpages
  {\ctxlua{thirddata.calendar.Photo_calendar_page(#1)}}
```

The cover page is created by means of a `\startstandardmakeup ... \stopstandardmakeup`:

```
\starttext

\startstandardmakeup[page=yes,doublesided=yes]
  \startplacefigure
    [location=middle,
     title=,
     number=]
    {\externalfigure[13][width=\paperwidth]}
  \stopplacefigure
  \midaligned{\bfc \labeltext{fotocal}~\getvariable{PocketDiary}{
    Year}}
\stopstandardmakeup

\Photocalendarpages{\getvariable{PocketDiary}{Year}}
\stoptext
```

The result looks as follows:

Title Page



January 2023

3. Menu Calendar

This project creates a calendar containing week tables. Each week is distributed over two facing pages. The first (odd) page should be filled with a picture in order to make sure that the week tables are shown on facing pages. The page header shows

the month and to the right the week number. The footer is showing the filename, page number and the year. In each day entry there is room to mention the meal to be cooked, and there is also space for adding comments.

This calendar might be used to observe a diet, plan meals or note the meals cooked to analyze how the different meals are distributed over time and what you eat most etc.

First the PocketDiary module and the collating marks module are loaded:

```
\usemodule[pocketdiary]
\usemodule[collatingmarks][Collatingmarks=yes]
```

PocketDiary uses Pagella as default font. If you want a different font, change it here:

```
\setupbodyfont[ibmplex,rm,8pt]
```

The main language is defined and the page numbering set to double-sided. The placement of the page number is handled with `\setupfootertexts[][]`:

```
\mainlanguage[de]
\setuppagenumbering[alternative=doublesided,location=]
```

This calendar will be typeset as A6 on A4 paper:

```
\setuppapersize[A6][A4]
```

The selected imposition scheme is “double-sided, four pages recto and four pages verso on two sheets of paper”. This results in a section of 16 pages.

```
\setuparranging[2*4*2]
```

If collating marks should be included then the module has to be setup as follows:

```
\setupMPvariables
   [pages=8,   % pages per sheet of paper doublesided
    sheets=2,  % sheets of paper used per section
    horpageshift=0mm]
```

A possible layout of the page looks as follows:

```
\setuplayout
  [topspace=.6cm,
   backspace=.6cm,
   header=2\bodyfontsize,
   headerdistance=.5\bodyfontsize,
   footer=1.2\bodyfontsize,
   footerdistance=.5\bodyfontsize,
   margin=0pt,
   height=middle,
   width=middle,
   location=middle]
```

From the PocketDiary variables we need only the year:

```
\setvariables
  [PocketDiary]
  [Year=2023]
```

The footer texts can be given individual content. In this case, because we have a multilingual interface, the `labeltexts` are used.

```
\setvariables
  [PocketDiaryFooter]
  [Lefttext=\labeltext{menu}-\labeltext{calendar},
   Centertext=\pagenumber,
   Righttext=\getvariable{PocketDiary}{Year}]
```

PocketDiary uses a blue separator line in the header and footer. You can change this here:

```
\setvariables
  [PocketDiaryColors]
  [Separatorline=blue,
   Gridline={s=.75}]
```

The setup of the week table is wrapped in `\startsetups ... \stopsetups`:

```
\startsetups table:week
   \setupTABLE[split=yes]
   \setupTABLE[c][1][width=.35\textwidth,frame=off]
   \setupTABLE[c][2][width=.65\textwidth,frame=off]
   \setupTABLE[r][height=2\lineheight,align=lohi]
   \setupTABLE[r][2,3,5,6,8,9,11,12,14,15,17,18,20,21][bottomframe
     =on]
\stopsetups
```

A possible multilingual interface could look like this:

```
\setuplabeltext[en][menu=Menu]
\setuplabeltext[en][comment=Comment]

\setuplabeltext[de][menu=Menü]
\setuplabeltext[de][comment=Kommentar]

\setuplabeltext[fr][menu=Menu]
\setuplabeltext[fr][comment=Commentaire]

\setuplabeltext[nl][menu=Menu]
\setuplabeltext[nl][comment=Commentaar]

\setuplabeltext[it][menu=Menu]
\setuplabeltext[it][comment=Commento]
```

```
\setuplabeltext[es][menu=Menú]
\setuplabeltext[es][comment=Comentario]
```

Most of the magic is done in Lua. This Lua block is called 52 times and delivers 52 week-tables:

```
\startluacode

  local report = logs.reporter("Menu calendar")

  function thirddata.calendar.menuweektable(mondaytimestamp,year,w
    )
    local stamp = mondaytimestamp
    local year = year
    local week = w
    local daynumber,dayname,monthname,holiday

    local day = stamp + 1 * 24 * 60 * 60

    report("Working in function: thirddata.calendar.menuweektable"
      )

    local monthname = string.lower(os.date("%B",day))

    --report("Month: %s Week: %s", monthname,week)

    thirddata.calendar.myheadertext(monthname,week)

    context.bTABLE({setups="table:week"})

    for i = 1,7 do
      local day = stamp + i * 24 * 60 * 60

     --report("Day-inloop: %s", day)

      daynumber   = tonumber(os.date("%d",day)) --day number
      dayname     = string.lower(os.date("%a",day)) -- day name
        short
      monthnumber = tonumber(os.date("%m",day)) -- month name
      holiday     = thirddata.calendar.checkchristianfeast(
        daynumber,
        monthnumber,year) -- Christian holiday

      if holiday ~= "" then
        report("Holiday: %s", holiday)
        report("Day number: %s", daynumber)
        report("Day name: %s",dayname)
      end

      context.bTR()
```

```
      context.bTD({style="bfa"})

        if dayname == "sun"  or holiday ~= "" and f ~= "ashw"
          then
          local dnu = daynumber
          local dna = dayname
          context.color({"red"},
            function() context(dnu) end)
          context("~")
          context.color({"red"},
            function() context.labeltext(dna) end)
        else
          context(daynumber)
          context("~")
          context.labeltext(dayname)
        end
      context.eTD()
      context.bTD()
        context.labeltext(holiday)
      context.eTD()
    context.eTR()
    context.bTR()
      context.bTD({nx="2"})
        context.labeltext("menu")
      context.eTD()
    context.eTR()
    context.bTR()
      context.bTD({nx="2"})
        context.labeltext("comment")
      context.eTD()
    context.eTR()
  end
  context.eTABLE()
  context.page({"yes"})
end

function thirddata.calendar.myheadertext(monthname,week)

  report("Working in function myheadertext")

  context.setupheadertexts{
    function()
      context("\\bgroup\\bfa")
      context.labeltext(monthname)
      context("\\egroup")
      context("\\hfill")
```

```
        context.labeltext("week")
        context(": ")
        context(week)
        return true
      end}
  end

  function thirddata.calendar.week(weeknumber,year)
     local w = weeknumber
     local mondaytimestamp = thirddata.calendar.weekcalendar(w,
       year)

     report("Working in function: thirddata.calendar.week")
     --report("Mondaytimestamp %s", mondaytimestamp)

     thirddata.calendar.menuweektable(mondaytimestamp,year,w)
  end

\stopluacode
```

In order to call the Lua code, we need a macro with two parameters, week number and year:

```
\define[2]\MenuWeektable{\ctxlua{thirddata.calendar.week(#1,#2)}}
```

If you use the collating marks module you must declare which parameters to use.

Each sheet of paper carries eight pages, we use two sheets of paper. Because we do not use a horizontal page-shift-list, the collating mark needs not to be shifted.

```
\setupMPvariables
 [pages=8,   % pages per sheet of paper doublesided
  sheets=2,  % sheets of paper used per section
  horpageshift=0mm]
```

Now we have everything prepared and can proceed to set up the document itself:

```
\starttext
  \startstandardmakeup[page=yes,doublsided=yes]
    \strut
    \godown[.3\textheight]
    \startalignment [middle]
      {\bfc \labeltext{menu}-\labeltext{calendar}}
      \vfill
      {\tfa \getvariable{PocketDiary}{Year}}
    \stopalignment
    \vfil
  \stopstandardmakeup

  \setupheadertexts[][]
```

```
  \strut\vfil

  \startplacefigure
    [number=,
     title=,
     location=middle]
     {\externalfigure[YourPicture.jpg][width=0.75\textwidth]}
  \stopplacefigure
  \vfill

  \page

  \setups{Footertext}

  \dorecurse{52}
    {\setvariables
      [PocketDiary]
      [Week=\recurselevel]
      \MenuWeektable{\recurselevel}
            {\getvariable{PocketDiary}{Year}}\page}

  \page

  \setupheadertexts[\bf \labeltext{notes}]

  \dorecurse
    {3}
    {\strut\page}
\stoptext
```

The result of this calendar pages look as follows:

4. Sun and Moon Data Calendar

Now, I was interested in getting a small calendar showing the sunrise and sunset times per day, including the moon phase. The machinery of the PocketDiary module provides the means to typeset this.

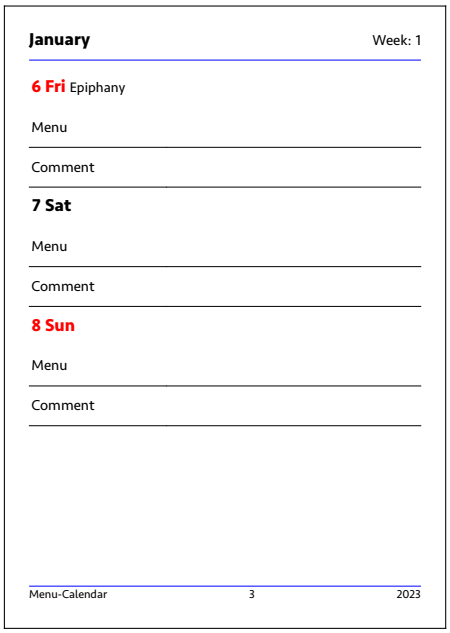
We load the module pocketdiary:

```
\usemodule[pocketdiary]
\usemodule[collatingmarks][Collatingmarks=yes]
```

This time we need to set the data for the PocketDiary and extend the first set with two fields ‘Place’ and ‘Country’:

```
\setvariables
  [PocketDiary]
  [Year=2023,
```

Even Page



Odd Page

```
   Place=Sambeek,
   Country=NL]
```

Setup for footer texts:

```
\setvariables
  [PocketDiaryFooter]
  [Lefttext={\jobname .pdf},
   Centertext=\pagenumber,
   Righttext=]
```

Changing the separator lines of header and footer:

```
\setvariables
  [PocketDiaryColors]
  [Separatorline=blue,
   Gridline={s=.75}]
```

The most important variable set is the one for the actual calculation of the required data:

```
\setvariables
  [PocketDiaryGeoPosition]
  [lat=51.63682, %Sambeek
   lon=5.96655,  %Sambeek
   timezoneoffset=1,
   continent=EU] %EU, US or empty for no DST
```

If the font needs to be different from the default PocketDiary font, set it up here:

```
\setupbodyfont[ibmplex,ss,10pt]
```

We setup the main language, the double-sided page numbering and switch off automatic page number placement, which is handled with `\setupfootertexts[][]` as well as the paper size:

```
\mainlanguage[en]
\setuppagenumbering[alternative=doublesided,location=]
\setuppapersize[A6][A4]
```

The layout is adapted to the intended book format:

```
\setuplayout
  [topspace=.6cm,
   backspace=.6cm,
   header=2\bodyfontsize,
   headerdistance=.5\bodyfontsize,
   footer=1.2\bodyfontsize,
   footerdistance=.5\bodyfontsize,
   margin=0pt,
   height=middle,
   width=88mm,
   location=middle]
```

The book will be printed in sections consisting of two sheets, each of them carrying 4 pages on recto and verso:

```
\setuparranging[2*4*2]
```

If the collating marks module is used we have to set it up:

```
\setupMPvariables % for collatingmarks
   [pages=8,  % pages per sheet of paper doublesided
    sheets=2,  % sheets of paper used per section
    horpageshift=0mm]
```

Observe the chunk of Lua that does all the magic; everything is set up as CLD (ConTEXt Lua Document):

```
\startluacode
local report = logs.reporter("Sundata calendar")

function thirddata.diary.year_of_monthtables(
  year,lat,lon,timeoffset,continent)

  report("Working in function: diary.year_of_monthtables")

  -- Determination of the DST start and stop date
  local dst_start,dst_stop = thirddata.diary.DST(year,continent)
```

```
  --report("DST start: %s, DST stop: %s",dst_start,dst_stop)

  for i= 1,1 do
    local monthname = string.lower(os.date
                      ("%B",os.time{year=year,month=i,day=1}))

    context.setupheadertexts(
      {
        function()
          context("\\bf")
          context.labeltext(monthname)
          return true
        end
      },
      {
        function()
          context(year)
          return true
        end
      },
      {
        function()
          context("\\bf")
          context.labeltext(monthname)
          return true
        end
      },
      {
        function()
          context(year)
          return true
        end
      })

    thirddata.diary.monthblock(i,year,lat,lon,
          timeoffset,dst_start,dst_stop)
  end
end

function thirddata.diary.monthblock(
  month,year,lat,lon,timeoffset,dst_start,dst_stop)

  report("Working in function diary.monthblock")

  -- returns number of weeks in a given month:
  local c = thirddata.calendar.month(month,year)
  -- returns the week number of the 1st day of a month:
```

```
local wkf = thirddata.calendar.weeknumber(1,month,year)

local nofdays = 1

if month == 2 and thirddata.calendar.isleapyear(year) ~= 365
  then
  nofdays = thirddata.calendar.nofdays_month(month,year) + 1
else
  nofdays = thirddata.calendar.nofdays_month(month,year)
end

--report("Number of days in month: %s", nofdays)
-- returns the week number of the last day of a month.

local wkl = thirddata.calendar.weeknumber(nofdays,month,year)

local monthname = thirddata.calendar.select_nameofmonth(month,
  year)

context.bTABLE({setups="table:monthblock,split=yes"})

  for i = 1,nofdays do

    local ordinalday  = thirddata.calendar.ordinalday(i,month,
      year)
    local weeknumber = thirddata.calendar.weeknumber(i,month,
      year)
    local holiday    = thirddata.calendar.checkchristianfeast(i,
      month,year)
    local moondata   = thirddata.moonphase.lunardays(year,month,
      i)
    local sunrise,sunset,lighthours =
            thirddata.srss.sundata(
              i,month,year,lat,lon,timeoffset,dst_start,dst_stop
                )

    context.bTR()
       context.bTD()

        local dayname = string.lower(os.date
                          ("%a",os.time{day=i,month=month,year=
                            year}))

        --report("Dayname: %s",dayname)
        context.starttabulate{"|lw(0.1\\textwidth)|lw(0.06\\textwidth)|l|
          "}
          context.NC()
            if  dayname == "sun"
```

```
                or holiday ~= "" and holiday ~= "asw" then
                local dnu = i
                local dna = dayname
                context.color({"red"},
                function() context(dnu) end)  --day number
                context("~")
                context.color({"red"},
                function() context.labeltext(dna) end)  --dayname
              else
                context(i)  --day number
                context("~")
                context.labeltext(dayname) --dayname
              end
            context.NC()
            context.switchtobodyfont{"6pt"}
              context.labeltext("week")
              context(": ")
              context(weeknumber)
            context.NC()
              context.switchtobodyfont{"6pt"}
              context.labeltext("day")
              context(": ")
              context(ordinalday)
            context.NC()
            context.NR()
          context.stoptabulate()
        context.eTD()
        context.bTD()
          context.starttabulate{"|lw(0.2\\textwidth)|l|l|l|l|"}
          context.NC()
            if holiday == "" then
              context.strut()
              context.NC()
              context.NC()
              context.NC()
              context.NC()
            else
              context.labeltext(holiday)
              context.NC()
              context.NC()
              context.NC()
              context.NC()
            end
            context.NR()
              context.NC()
```

```
                if tonumber(moondata) then
                  context.labeltext("moondays")
                else
                  context.labeltext("moon")
                end
                  context(":~")
                  context(moondata)
                context.NC()
                  --context.labeltext("sunrise")
                  context("\\SunA[background=Sunrise]{}")
                  context("\\quad :~")
                  context(sunrise)
                context.NC()
                  --context.labeltext("sunset")
                  context("\\SunA[background=Sunset]{}")
                  context("\\quad :~")
                  context(sunset)
                context.NC()
                  --context.labeltext("lighthours")
                  context("\\SunB[background=Light]{}")
                  context("~:~")
                  context(lighthours)
                context.NC()
              context.NR()
            context.stoptabulate()
          context.eTD()
        context.eTR()
      end
  context.eTABLE()
  context.page()
end

\stopluacode
```

The table layout is wrapped in a `\startsetups ... \stopsetups` environment:

```
\startsetups[table:monthblock]
  \setupTABLE[split=yes]
  \setupTABLE[each][each][frame=on]
  \setupTABLE[r][each][offset=0pt,loffset=3pt,roffset=3pt,align=
    lohi]
  \setupTABLE[c][1][width=0.35\textwidth,align=lohi]
  \setupTABLE[c][2][style={\switchtobodyfont[6pt]},width=0.65
    \textwidth] \stopsetups
```

We define a macro with 5 parameters to call the Lua function:

```
\define[5]\Yearofmonthtables
  {\ctxlua{thirddata.diary.year_of_monthtables(#1,#2,#3,#4,#5)}}
```

The macro with the collected information is put into a buffer:

```
\startbuffer[Yearofmonthtables]
  \Yearofmonthtables %Year,lat,lon,timeoffset,continent
     {\getvariable{PocketDiary}{Year}}
     {\getvariable{PocketDiaryGeoPosition}{lat}}
     {\getvariable{PocketDiaryGeoPosition}{lon}}
     {\getvariable{PocketDiaryGeoPosition}{timeoffset}}
     {"\getvariable{PocketDiaryGeoPosition}{continent}"}
\stopbuffer
```

For the setup of the textual parts on the title page, and the editorial, we add some label texts for international interfacing:

```
\setuplabeltext[en][title=Title]
\setuplabeltext[nl][title=Titel]
\setuplabeltext[de][title=Titel]
\setuplabeltext[fr][title=Titre]
\setuplabeltext[it][title=Titolo]
\setuplabeltext[es][title=Título]

\setuplabeltext[en][author=Author]
\setuplabeltext[nl][author=Auteur]
\setuplabeltext[de][author=Autor]
\setuplabeltext[fr][author=Auteur]
\setuplabeltext[it][author=autore]
\setuplabeltext[es][author=Autor]

\setuplabeltext[en][typography=Typography]
\setuplabeltext[nl][typography=Typografie]
\setuplabeltext[de][typography=Typographie]
\setuplabeltext[fr][typography=Typographie]
\setuplabeltext[it][typography=Tipografia]
\setuplabeltext[es][typography=Tipografía]

\setuplabeltext[en][compilation=Compilation date]
\setuplabeltext[nl][compilation=Compilatie datum]
\setuplabeltext[de][compilation=Kompilationsdatum]
\setuplabeltext[fr][compilation=Date de compilation]
\setuplabeltext[it][compilation=Dato di compilazione]
\setuplabeltext[es][compilation=Fecha de compilación]
```

Now we have everything to start typesetting the document:

```
\starttext
```

```
\startstandardmakeup[page=yes,doublesided=no]
  \strut
  \startalignment [middle]
     \godown[.3\textheight]
     \startmode[*en]
       {\bfc \labeltext{sunstar}- and
         \labeltext{moon}-\labeltext{calendar}}
     \stopmode
     \startmode[*nl]
       {\bfc \labeltext{sunstar}- en
         \labeltext{moon}-\labeltext{calendar}}
     \stopmode
     \startmode[*de]
       {\bfc \labeltext{sunstar}n- und
         \labeltext{moon}-\labeltext{calendar} }
     \stopmode
     \startmode[*fr]
       {\bfc \labeltext{calendar} du
         \labeltext{sunstar} et de la \labeltext{moon}}
     \stopmode
     \startmode[*it]
       {\bfc \labeltext{calendar} di
         \labeltext{sunstar} e della \labeltext{moon}}
     \stopmode
     \startmode[*es]
       {\labeltext{calendar} del
         \labeltext{sunstar} y la \labeltext{moon}}
     \stopmode
     \blank
     {\bfb \getvariable{PocketDiary}{Place}}
     \vfill
     {\tfa \getvariable{PocketDiary}{Year}}
  \stopalignment
  \vfil
 \stopstandardmakeup

 \startstandardmakeup[page=yes,doublesided=no]
   \starttabulate[|l|p|]
      \NC \labeltext{title}
     \EQ
       \startmode[*en]
         {\labeltext{sunstar}- and
           \labeltext{moon}-\labeltext{calendar}}
       \stopmode
       \startmode[*nl]
```

```
        {\labeltext{sunstar}- en
          \labeltext{moon}-\labeltext{calendar}}
      \stopmode
      \startmode[*de]
        {\labeltext{sunstar}n- und
          \labeltext{moon}-\labeltext{calendar} }
      \stopmode
      \startmode[*fr]
        {\labeltext{calendar} du
          \labeltext{sunstar} et de la \labeltext{moon}}
      \stopmode
      \startmode[*it]
        {\labeltext{calendar} di
          \labeltext{sunstar} e della \labeltext{moon}}
      \stopmode
      \startmode[*es]
        {\labeltext{calendar} del
          \labeltext{sunstar} y la \labeltext{moon}}
      \stopmode
    \NC\NR
    \NC \labeltext{author}
    \EQ \getvariable{PocketDiaryAddress}{Forename}~
        \getvariable{PocketDiaryAddress}{Familyname} \NC\NR
    \NC \labeltext{typography} \EQ \CONTEXT\ with \LUAMETATEX
      \NC\NR
    \NC \labeltext[compilation] \EQ \currentdate \NC\NR
  \stoptabulate
  \vfil
  \startmode[*en]
    This calendar is generated for \getvariable{PocketDiary}{
      Place},
      \getvariable{PocketDiary}{Country}
  \stopmode
  \startmode[*de]
    Dieser Kalender ist berechnet für \getvariable{PocketDiary}{
      Place},
      \getvariable{PocketDiary}{Country}
  \stopmode
  \startmode[*nl]
    Deze kalender is opgemaakt voor \getvariable{PocketDiary}{
      Place},
      \getvariable{PocketDiary}{Country}
  \stopmode
  \startmode[*fr]
```

```
      Ce calendrier est fait pour \getvariable{PocketDiary}{Place}
        ,
        \getvariable{PocketDiary}{Country}
    \stopmode
    \startmode[*it]
       Questo calendario è prodotto per
              \getvariable{PocketDiary}{Place},
         \getvariable{PocketDiary}{Country}
    \stopmode
    \startmode[*es]
      Este calendario está calculado para
             \getvariable{PocketDiary}{Place},
        \getvariable{PocketDiary}{Country}
    \stopmode

    \starttabulate[|l|r|]
      \NC Longitude
      \EQ \getvariable{PocketDiaryGeoPosition}{lon} \NC\NR
      \NC Latitude
      \EQ  \getvariable{PocketDiaryGeoPosition}{lat} \NC\NR
    \stoptabulate
  \stopstandardmakeup

  \getbuffer[Yearofmonthtables]

\stoptext
```

The resulting pages look as follows:

5. Calendar A5 one Page per Day

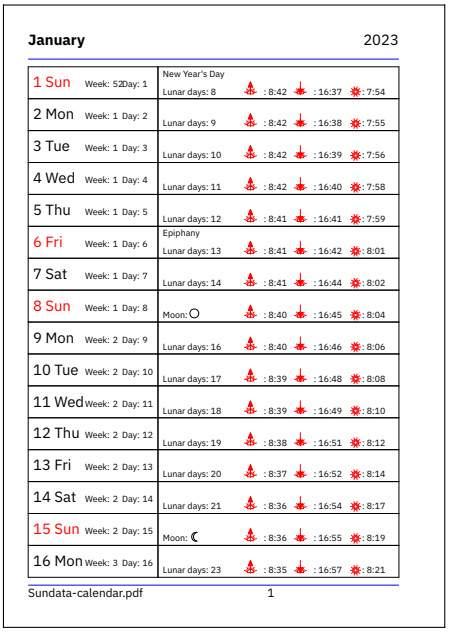
Knowing that the PocketDiary has very small pages and it covers 1 week only, it is desirable to be able to produce a calendar which shows each day on a single page. The following example does just this.

We load the module `\t-pocketdiary.tex` and the collating marks module:

```
\usemodule[pocketdiary]
\usemodule[collatingmarks][Collatingmarks=yes]
```

The international interface is enabled through modes:

```
\definemode[English,Dutch,German,French,Italian,Spanish][keep]

\enablemode[English]
% \enablemode[German]
% \enablemode[Dutch]
% \enablemode[Italian]
```

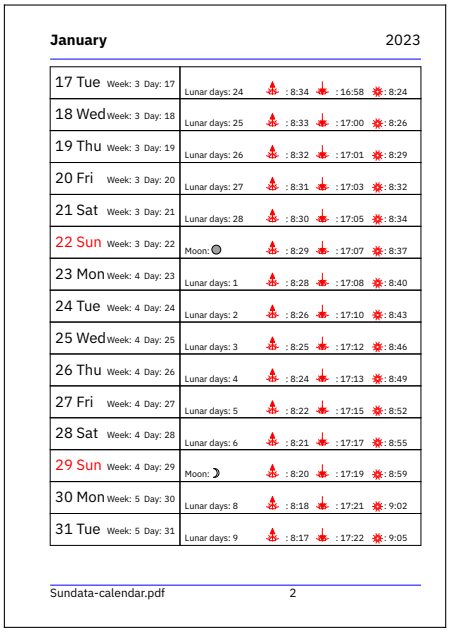


January 2023

1 Sun	Week: 52 Day: 1	New Year's Day Lunar days: 8 : 8:42 : 16:37 : 7:54
2 Mon	Week: 1 Day: 2	Lunar days: 9 : 8:42 : 16:38 : 7:55
3 Tue	Week: 1 Day: 3	Lunar days: 10 : 8:42 : 16:39 : 7:56
4 Wed	Week: 1 Day: 4	Lunar days: 11 : 8:42 : 16:40 : 7:58
5 Thu	Week: 1 Day: 5	Lunar days: 12 : 8:41 : 16:41 : 7:59
6 Fri	Week: 1 Day: 6	Epiphany Lunar days: 13 : 8:41 : 16:42 : 8:01
7 Sat	Week: 1 Day: 7	Lunar days: 14 : 8:41 : 16:44 : 8:02
8 Sun	Week: 1 Day: 8	Moon: ○ : 8:40 : 16:45 : 8:04
9 Mon	Week: 2 Day: 9	Lunar days: 16 : 8:40 : 16:46 : 8:06
10 Tue	Week: 2 Day: 10	Lunar days: 17 : 8:39 : 16:48 : 8:08
11 Wed	Week: 2 Day: 11	Lunar days: 18 : 8:39 : 16:49 : 8:10
12 Thu	Week: 2 Day: 12	Lunar days: 19 : 8:38 : 16:51 : 8:12
13 Fri	Week: 2 Day: 13	Lunar days: 20 : 8:37 : 16:52 : 8:14
14 Sat	Week: 2 Day: 14	Lunar days: 21 : 8:36 : 16:54 : 8:17
15 Sun	Week: 2 Day: 15	Moon: ☾ : 8:36 : 16:55 : 8:19
16 Mon	Week: 3 Day: 16	Lunar days: 23 : 8:35 : 16:57 : 8:21

Sundata-calendar.pdf 1

Odd Page



January 2023

17 Tue	Week: 3 Day: 17	Lunar days: 24 : 8:34 : 16:58 : 8:24
18 Wed	Week: 3 Day: 18	Lunar days: 25 : 8:33 : 17:00 : 8:26
19 Thu	Week: 3 Day: 19	Lunar days: 26 : 8:32 : 17:01 : 8:29
20 Fri	Week: 3 Day: 20	Lunar days: 27 : 8:31 : 17:03 : 8:32
21 Sat	Week: 3 Day: 21	Lunar days: 28 : 8:30 : 17:05 : 8:34
22 Sun	Week: 3 Day: 22	Moon: ● : 8:29 : 17:07 : 8:37
23 Mon	Week: 4 Day: 23	Lunar days: 1 : 8:28 : 17:08 : 8:40
24 Tue	Week: 4 Day: 24	Lunar days: 2 : 8:26 : 17:10 : 8:43
25 Wed	Week: 4 Day: 25	Lunar days: 3 : 8:25 : 17:12 : 8:46
26 Thu	Week: 4 Day: 26	Lunar days: 4 : 8:24 : 17:13 : 8:49
27 Fri	Week: 4 Day: 27	Lunar days: 5 : 8:22 : 17:15 : 8:52
28 Sat	Week: 4 Day: 28	Lunar days: 6 : 8:21 : 17:17 : 8:55
29 Sun	Week: 4 Day: 29	Moon: ☽ : 8:20 : 17:19 : 8:59
30 Mon	Week: 5 Day: 30	Lunar days: 8 : 8:18 : 17:21 : 9:02
31 Tue	Week: 5 Day: 31	Lunar days: 9 : 8:17 : 17:22 : 9:05

Sundata-calendar.pdf 2

Even Page

January 2023

```
% \enablemode[French]
% \enablemode[Spanish]
```

A couple of variables have to be set for the PocketDiary module:

```
\setvariables
  [PocketDiary]
  [Year=2023,
   Week=1,
   Day=1,
   Month=1,
   Nextyear=yes,
   Yearnext=2024]

\setvariables
  [PocketDiaryFooter]
  [Lefttext={\jobname .pdf},
   Centertext=\pagenumber,
   Righttext=]

\setvariables
  [PocketDiaryColors]
  [Separatorline=blue,
   Gridline={s=.75}]

\setvariables %Koziegłowy 52.467860 16.981240
```

```
  [PocketDiaryGeoPosition]
  [lat=52.467860,
   lon=16.981240,
   timezoneoffset=1,
   continent=EU] % EU, US or empty for no Daylight Saving Time
     (DST)

\setvariables
  [PocketDiaryAddress]
  [Familyname=YourFamilyName,
   Forename=YourName,
   Street=YourStreet and Number,
   Zipcode=YourPostalCode,
   City=YourPlace,
   Country=YourCountry,
   Phone=YourPhoneNumber,
   Mobile=YourMobilePhoneNumber,
   E-mail=YourEmail@YourProvider,
   Web=YourWWW]
```

If we want another font to be used instead of the preset Pagella, then we set up our font. It is a good idea to also add a fontfeature which enables tabular numbers:

```
\setupbodyfont[ibmplex,ss,10pt]

\definefontfeature[f:Tabular][tnum=yes]

\addfeature[f:Tabular]
```

For enabling the international interfaces for each defined mode the mainlanguage is set.

```
\startmode[*en]
  \mainlanguage[en]
\stopmode

\startmode[*de]
  \mainlanguage[de]
\stopmode

\startmode[*fr]
  \mainlanguage[fr]
\stopmode

\startmode[*it]
  \mainlanguage[it]
\stopmode

\startmode[*es]
```

```
  \mainlanguage[es]
\stopmode
```

The document is double sided printed, the page number will be set in the `\setupfootertexts`:

```
\setuppagenumbering[alternative=doublesided,location=]
```

We use A4 paper landscape to typset A5 sized pages:

```
\setuppapersize[A5][A4,landscape]
```

We adapt the page layout to the size of the page:

```
\setuplayout
  [topspace=.6cm,
   backspace=1.5cm,
   header=2\bodyfontsize,
   headerdistance=.5\bodyfontsize,
   footer=1.2\bodyfontsize,
   footerdistance=.5\bodyfontsize,
   margin=0pt,
   height=middle,
   width=middle,
   location=middle]
```

We enable the imposition: 2 pages recto and verso using 4 sheets of A4 to form one section

```
\setuparranging[2*2*4]
```

The setup of the Collating marks module looks as follows:

```
\setupMPvariables % for collatingmarks
  [pages=8,  % pages per sheet of paper doublesided
   sheets=2,  % sheets of paper used per section
   horpageshift=-0.5mm]
```

Now we prepare the lua code. The document is made up as a ConTEXt-lua-document:

```
\startluacode

  local report = logs.reporter("Calendar-1-day-per-page")

  function thirddata.diary.year_of_monthtables(year,lat,lon,
      timeoffset,continent)

    report("Working in function: diary.year_of_monthtables")

    for i = 1,12 do
      local monthname = string.lower(os.date
                        ("%B",os.time{year=year,month=i,day=1}))
```

```
    thirddata.diary.header_text(monthname,year)

    thirddata.diary.monthblock(i,year,lat,lon,
            timeoffset,continent)
  end
end

function thirddata.diary.monthblock(month,year,lat,lon,
  timeoffset,continent)

  report("Working in function diary.monthblock")

  local nofdays = 1

  local monthname = thirddata.calendar.select_nameofmonth(month,
    year)
  local yeardays   = thirddata.calendar.isleapyear(year)

  if month == 2 and yeardays == 366 then
    nofdays = thirddata.calendar.nofdays_month(month,year) + 1
  else
    nofdays = thirddata.calendar.nofdays_month(month,year)
  end

  --report("Number of days in month: %s", nofdays)

  for i = 1,nofdays do
    local ordinalday = thirddata.calendar.ordinalday(i,month,
      year)
    local weeknumber = thirddata.calendar.weeknumber(i,month,
      year)
    local dayname    = string.lower(os.date
      "%a",os.time{day=i,month=month,year=year}))
    local holiday    =
      thirddata.calendar.checkchristianfeast(i,month,year)
    local moondata   = thirddata.moonphase.lunardays(year,month,
      i)
    local dst_start,dst_stop = thirddata.diary.DST(year,
      continent)
    local sunrise,sunset,lighthours =
            thirddata.srss.sundata(i,month,year,lat,lon,
              timeoffset,dst_start,dst_stop)

    if holiday ~= "" then
      report("Holiday: %s", holiday)
      report("Day number: %s", i)
    end
```

```
      context.bTABLE({setups="table:dayblock"})
        context.bTR()
          context.bTD()
            context.starttabulate
                {"|lw(0.1\\textwidth)|p(0.06\\textwidth)|p|"}
              context.NC()
                context("\\bf")
                --report("Daynumber: %s Month %s", i,month)
                if  dayname == "sun"
                  or holiday ~= "" and holiday ~= "ashw" then
                  local dnu = i
                  local dna = dayname
                  context.color({"red"},
                    function() context(dnu) end)  --day number
                  context("~")
                  context.color({"red"},
                    function() context.labeltext(dna) end)
                      --dayname
                else
                  context(i)  --day
                  context("~")
                  context.labeltext(dayname) --dayname
                end
              context.NC()
                context.switchtobodyfont{"6pt"}
                context.labeltext("week")
                context(":")
                context.par()
                context(weeknumber)
              context.NC()
                context.switchtobodyfont{"6pt"}
                context.labeltext("day")
                context(":")
                context.par()
                context(ordinalday)
                context("/")
                context(yeardays)
              context.NC()
              context.NR()
            context.stoptabulate()
          context.eTD()
          context.bTD()
            context.starttabulate{"|lw(0.2\\textwidth)|p|p|p|p|"}
              context.NC()
                if holiday == "" then
```

```
                context.strut()
              else
                context.labeltext(holiday)
              end
            context.NC()
              if tonumber(moondata) then
                context.labeltext("moondays")
              else
                context.labeltext("moon")
              end
              context(":~")
              context.par()
              context(moondata)
            context.NC()
              context("\\SunA[background=Sunrise]{}")
              context.par()
              context(sunrise)
            context.NC()
              context("\\SunA[background=Sunset]{}")
              context.par()
              context(sunset)
            context.NC()
              context("\\SunB[background=Light]{}")
              context.par()
              context(lighthours)
            context.NC()
            context.NR()
          context.stoptabulate()
        context.eTD()
      context.eTR()
    context.eTABLE()
    context.page()
  end
end

function thirddata.diary.header_text(monthname,year)
  context.setupheadertexts(
    {function()
       context("\\bf")
       context.labeltext(monthname)
       return true
     end
    },
    {function()
       context(year)
```

```
          return true
        end
      },
      {function()
          context("\\bf")
          context.labeltext(monthname)
          return true
        end
      },
      {function()
          context(year)
          return true
        end
      })
  end

\stopluacode
```

In order to have a table layout according to our wishes, we wrap the setups of the ‘dayblock’ in a setups block:

```
\startsetups[table:dayblock]
  \setupTABLE[each][each][frame=off]
  \setupTABLE[r][each]
    [offset=0pt,loffset=3pt,roffset=3pt,align=lohi]
  \setupTABLE[c][1]
    [width=0.35\textwidth,align=lohi]
  \setupTABLE[c][2]
    [style={\switchtobodyfont[6pt]},width=0.65\textwidth]
      \stopsetups
```

Of course, we need a macro to be able to call the functions in the lua-code. The macro has 5 parameters:

```
\define[5]\Yearofmonthtables{%
  \ctxlua{thirddata.diary.year_of_monthtables(#1,#2,#3,#4,#5)}}
```

I use a buffer containing the gathered parameters for the above defined macro:

```
\startbuffer[Yearofmonthtables]
  \Yearofmonthtables %Year,lat,lon,timeoffset,continent
     {\getvariable{PocketDiary}{Year}}
     {\getvariable{PocketDiaryGeoPosition}{lat}}
     {\getvariable{PocketDiaryGeoPosition}{lon}}
     {\getvariable{PocketDiaryGeoPosition}{timeoffset}}
     {"\getvariable{PocketDiaryGeoPosition}{continent}"}
\stopbuffer
```

The page carries a page background. This background shows a time scale on the left starting at 6:00 and ending at 22:00 in hourly steps. Each hour is separated by a grey line. The background is a MetaFundrawing. There is then an overlay accepting the background.

```
\startreusableMPgraphic{Time-Lines}
  numeric w; w:= OverlaytWidth;
  numeric h; h:= (OverlayHeight-2cm)/18;
  numeric t;
  path p,q;
  p := unitsquare xscaled OverlayWidth yscaled (OverlayHeight-2cm)
    ;

  q:= llcorner p -- lrcorner p;
  draw q withpen pencircle scaled .5pt
      withcolor \MPcolor{Grid};
  for i = 1 upto 17:
    draw q  shifted (0,i*h) withpen pencircle scaled .5pt
      withcolor \MPcolor{Grid};
      t :=   24 - 1 - i ;
      label.top(textext(""& decimal t & ":00"),point 0 of q)
        shifted (15pt,((i-1)*h)+1pt);
  endfor;
\stopreusableMPgraphic

\defineoverlay[Textbackground][\reuseMPgraphic{Time-Lines}]
```

Fo the editorial some label texts are setup for the international interface:

```
\setuplabeltext[en][title=Title]
\setuplabeltext[nl][title=Titel]
\setuplabeltext[de][title=Titel]
\setuplabeltext[fr][title=Titre]
\setuplabeltext[it][title=Titolo]
\setuplabeltext[es][title=Título]

\setuplabeltext[en][author=Author]
\setuplabeltext[nl][author=Auteur]
\setuplabeltext[de][author=Autor]
\setuplabeltext[fr][author=Auteur]
\setuplabeltext[it][author=autore]
\setuplabeltext[es][author=Autor]

\setuplabeltext[en][typography=Typography]
\setuplabeltext[nl][typography=Typografie]
\setuplabeltext[de][typography=Typographie]
\setuplabeltext[fr][typography=Typographie]
\setuplabeltext[it][typography=Tipografia]
```

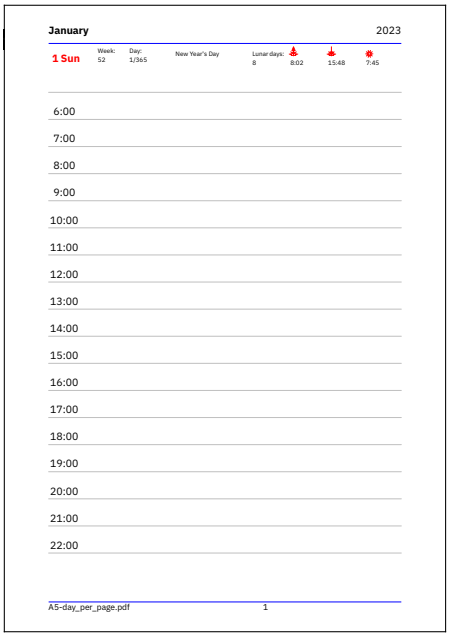
```
\setuplabeltext[es][typography=Tipografía]

\setuplabeltext[en][compilation=Compilation date]
\setuplabeltext[nl][compilation=Compilatie datum]
\setuplabeltext[de][compilation=Kompilationsdatum]
\setuplabeltext[fr][compilation=Date de compilation]
\setuplabeltext[it][compilation=Dato di compilazione]
\setuplabeltext[es][compilation=Fecha de compilación]
```

Before we can run the compilation we need to set the document up. In order to have a full section at the end, I added some todo and grid lined pages (borrowed from the PocketDiary module) My setup looks as follows:

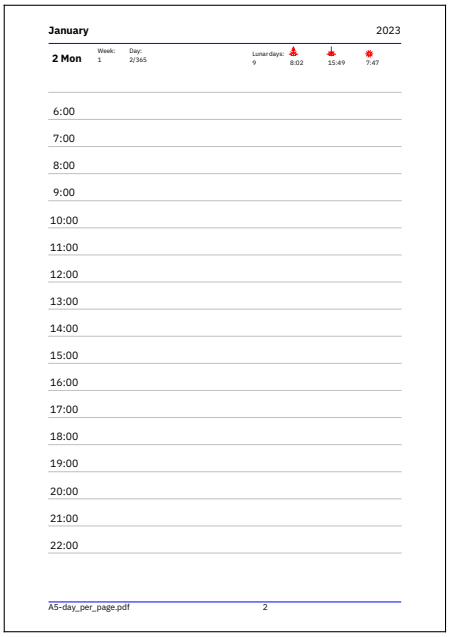
```
\starttext

  \startstandardmakeup[page=yes,doublesided=no]
    \strut
    \startalignment [middle]
      \godown[.3\textheight]
      {\bfa \labeltext{calendar}}
      \vfil
      {\bf \getvariable{PocketDiary}{Year}}
    \stopalignment
  \stopstandardmakeup

  \startstandardmakeup[page=yes,doublesided=no]
    \starttabulate[|l|p|]
      \NC \labeltext{title}
      \EQ {\bf \labeltext{calendar}} \NC\NR
      \NC \labeltext{author}
      \EQ \getvariable{PocketDiaryAddress}{Forename}
          \getvariable{PocketDiaryAddress}{Familyname} \NC\NR
      \NC \labeltext{typography}
      \EQ \CONTEXT\ with \LUAMETATEX \NC\NR
      \NC \labeltext{compilation}
      \EQ \currentdate \NC\NR
    \stoptabulate
  \stopstandardmakeup

  \getbuffer[Lost-Returnto]

  \setupbackgrounds[text][background=]

  \setupbackgrounds[text][background=Textbackground]

  \getbuffer[Yearofmonthtables]

  \setupbackgrounds[text][background=]
```

```
  \getbuffer[Yearcurrentplan]

  \setupheadertexts[\bfa \getvariable{PocketDiary}{Yearnext}]

  \getbuffer[Yearnextplan]

  \dorecurse
    {3}
    {\getbuffer[Todo]}

  \dorecurse
    {10}
    {\getbuffer[Lines]}

\stoptext
```

After running this file we get 12 sections per four sheets of A4 landscape paper with two pages on the recto and verso side. Each first sheet of the set of four carries a collating mark:



Odd Page



Even Page

Calendar with One Day per Page

6. Calendar A6 or A5 two Facing Pages per Week

The A5 version with one day per page is a rather big book. For those who need a smaller diary the following example can be an option. It can be typeset as a A6 booklet or A5 book. This document presents a whole week on two facing pages in

the form of a table. Each working day is a separate row. The weekend is shown in a single row.

As in the previous examples we need to load the PocketDiary module. Because this book has multiple section we also load the collating marks module:

```
\usemodule[pocketdiary]
\usemodule[collatingmarks][Collatingmarks=yes]
```

We will use different modes for the international interface and two additional modes to be able to switch between an A6 and an A5 version of the calendar.

```
\definemode[English,Dutch,German,French,Italian,Spanish][keep]
\definemode[A5,A6][keep]

\enablemode[English]
% \enablemode[German]
% \enablemode[Dutch]
% \enablemode[Italian]
% \enablemode[French]
% \enablemode[Spanish]

\enablemode[A6]
% \enablemode[A5]
```

Of course we need to tell the PocketDiary module wat we want and how the page should look and we have to give geo-information too:

```
\setvariables
  [PocketDiary]
  [Year=2024,
   Week=1,
   Day=1,
   Month=1,
   Nextyear=yes]

\setvariables
  [PocketDiaryFooter]
  [Lefttext={\jobname .pdf},
   Centertext=\pagenumber,
   Righttext=]

\setvariables
  [PocketDiaryColors]
  [Separatorline=blue,
   Gridline={s=.75}]

\setvariables %Koziegłowy 52.467860 16.981240
  [PocketDiaryGeoPosition]
```

```
  [lat=52.467860,
   lon=16.981240,
   timezoneoffset=1,
   continent=EU] % Eu, US or empty for no Daylight Saving Time
     (DST)

\setvariables
  [PocketDiaryAddress]
  [Familyname=YourFamilyName,
   Forename=YourName,
   Street=YourStreet and Number,
   Zipcode=YourPostalCode,
   City=YourPlace,
   Country=YourCountry,
   Phone=YourPhoneNumber,
   Mobile=YourMobilePhoneNumber,
   E-mail=YourEmail@YourProvider,
   Web=YourWWW]
```

Without setting up the bodyfont, PocketDiary will use Pagella. Because we will typset a lot of numbers, it is advisable to enable tabular figures for the selected font.

```
\setupbodyfont[ibmplex,ss,10pt]

\definefontfeature[f:Tabular][tnum=yes]

\addfeature[f:Tabular]
```

Through modes for each language we set the mainlanguage:

```
\startmode[*en]
  \mainlanguage[en]
\stopmode

\startmode[*de]
  \mainlanguage[de]
\stopmode

\startmode[*fr]
  \mainlanguage[fr]
\stopmode

\startmode[*it]
  \mainlanguage[it]
\stopmode

\startmode[*es]
  \mainlanguage[es]
\stopmode
```

We want to have a double-sided printed document. The page numbers will be set up in the footer text. We also adapt the layout of the page to our needs:

```
\setuppagenumbering[alternative=doublesided,location=]

\setuplayout
  [topspace=.6cm,
   backspace=1cm,
   header=2\bodyfontsize,
   headerdistance=.5\bodyfontsize,
   footer=1.2\bodyfontsize,
   footerdistance=.5\bodyfontsize,
   margin=0pt,
   height=middle,
   width=middle,
   location=middle]
```

Again we use the mode mechanism to be able to switch between A6 and A5 sized output. In the A6 version we use also a vertical shiftlist. This provides us with the possibilty of making a head-cut of 2.5 mm of the book block.

```
\startmode[A6]
  \definepapersize[Agenda][width=100mm,height=140mm]

  \setuppapersize[Agenda][A4,portrait]

  \definepageshift
    [ver]
    [vertical]
    [2.5mm,2.5mm,2.5mm,2.5mm,2.5mm,2.5mm,2.5mm,2.5mm,
     2.5mm,2.5mm,2.5mm,2.5mm,2.5mm,2.5mm,2.5mm,2.5mm]

  \setuppageshift[paper][ver]

  \setuparranging[2*4*2]

  \setupMPvariables % for collatingmarks
    [pages=8,  % pages per sheet of paper doublesided
     sheets=2,  % sheets of paper used per section
     horpageshift=-0.5mm]
\stopmode

\startmode[A5]
  \definepapersize[Agenda][width=148mm,height=210mm]

  \setuppapersize[Agenda][A4,landscape]

  \setuparranging[2*2*4]
```

```
  \setupMPvariables % for collatingmarks
    [pages=4,  % pages per sheet of paper doublesided
     sheets=4,  % sheets of paper used per section
     horpageshift=0mm]
\stopmode
```

This example is again set up as a ConTEXt-lua-document. The lua-code is as follows:

```
\startluacode

  local report = logs.reporter("Calendar-1-week-per-2-pages")

  function thirddata.diary.year_of_weektables(year,
              lat,lon,timeoffset,continent)

    local year       = year
    local lat        = lat
    local lon        = lon
    local timeoffset = timeoffset
    local continent  = continent

    report("Working in function: diary.year_of_weektables")

    for i = 1,12 do  --i is month
      thirddata.diary.monthblock(i,year,lat,lon,timeoffset,
        continent)
    end
  end

  function thirddata.diary.monthblock(month,year,
              lat,lon,timeoffset,continent)

    report("Working in function diary.monthblock: %s", month)

        --return number of weeks in a given month
    local c = thirddata.calendar.month(month,year)
        -- return weeknumber of 1st day of a given month
    local wkf = thirddata.calendar.weeknumber(1,month,year)
    -- last week of a given month based on wkf and c, is wrong for
      January
    local wkl = wkf + c - 1

    --report("Weeks in the selected month: %s", c)
    --report("First week of month: %s", wkf)

    if month == 1 and wkf > 51 then

      --report("Calculating in week number %s", wkf)
```

```
      thirddata.diary.setup_header_texts(wkf,month,year)
      year = year - 1
      thirddata.diary.thisweek(wkf,year,lat,lon,timeoffset,
        continent)
      year = year + 1
      wkf = 1
      wkl = wkf + c - 2

      for i = wkf, wkl do
        thirddata.diary.setup_header_texts(i,month,year)
        thirddata.diary.thisweek(i,year,lat,lon,timeoffset,
          continent)
      end
    else
      for i = wkf, wkl do

        --report("Calculating in week number %s", i)

        thirddata.diary.setup_header_texts(i,month,year)
        thirddata.diary.thisweek(i,year,
          lat,lon,timeoffset,continent)
       end
    end
  end

  function thirddata.diary.setup_header_texts(week,month,year)

    local monthname = string.lower(os.date
            ("%B",os.time{year=year,month=month,day=1}))

    context.setupheadertexts(
      {function()
         context("\\bf")
         context.labeltext(monthname)
         return true
       end
      },
      {function()
         context.labeltext("week")
         context(": ~")
         context(week)
         context("\\quad")
         context(year)
         return true
       end
      },
      {function()
```

```
          context("\\bf")
          context.labeltext(monthname)
          return true
        end
      },
      {function()
          context.labeltext("week")
          context(": ~")
          context(week)
          context("\\quad")
          context(year)
          return true
        end
      })
  end

\stopluacode
```

The week table setups are wrapped in a `\startsetups ... \stopsetups`:

```
\startsetups[table:week]
  \setupTABLE[split=yes]
  \setupTABLE[each][each][frame=on]
  \setupTABLE
    [r][each]
    [offset=0pt,loffset=3pt,roffset=3pt,
     height=0.23\textheight]
  \setupTABLE[c][1][width=0.5\textwidth]
  \setupTABLE[c][2][width=0.5\textwidth]
\stopsetups
```

This time the macro for calling the lua-code needs five parameters again:

AS in previous examples I use a buffer for the macro with its gathered parameters:

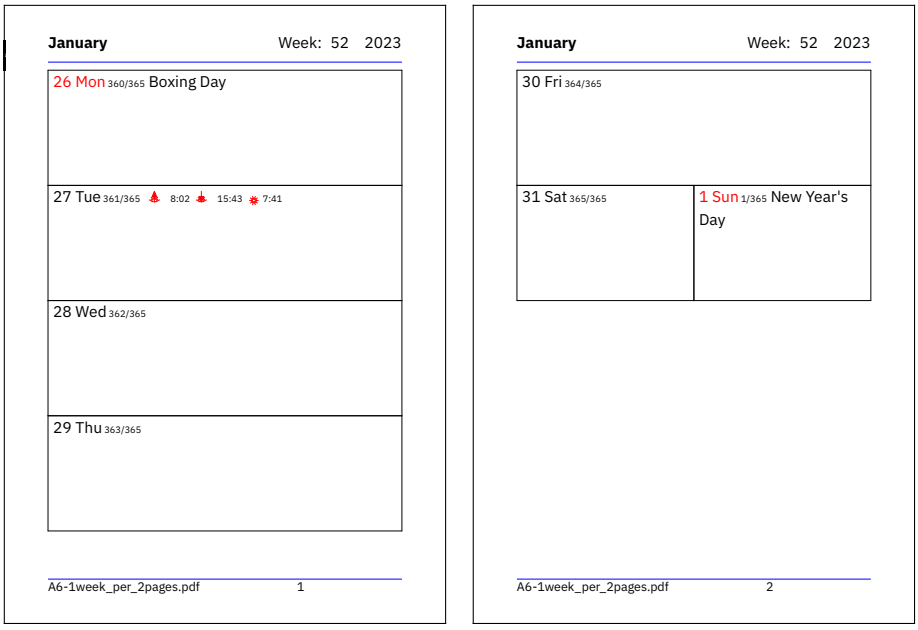
```
\startbuffer[Yearofweektables]
  \Yearofweektables %Year,lat,lon,timeoffset,continent
     {\getvariable{PocketDiary}{Year}}
     {\getvariable{PocketDiaryGeoPosition}{lat}}
     {\getvariable{PocketDiaryGeoPosition}{lon}}
     {\getvariable{PocketDiaryGeoPosition}{timeoffset}}
     {"\getvariable{PocketDiaryGeoPosition}{continent}"}
\stopbuffer
```

Finally again we assemble the document's layout. This time the international interfacing for the editorial is based on the `\translate[...]` command. In order to have a full section at the end, a couple of grid-line sheets (template included in PocketDiary) is added:

```
\starttext

  \startstandardmakeup[page=yes,doublesided=no]
    \strut
    \godown[.3\textheight]
    \startalignment [middle]
      {\bfa \labeltext{calendar}}
      \vfil
      {\bf \getvariable{PocketDiary}{Year}}
    \stopalignment
  \stopstandardmakeup

  \startstandardmakeup[page=yes,doublesided=no]
    \starttabulate[|l|p|]
       \NC \translate[en=Title,
                      nl=Titel,
                      de=Titel,
                      fr=Titre,
                      it=Titolo]
      \EQ {\bf \labeltext{calendar}} \NC\NR
      \NC \translate[en=Author,
                      nl=Auteur,
                      de=Autor,
                      fr=Auteur,
                      it=autore,
                      es=Autor]
      \EQ \getvariable{PocketDiaryAddress}{Forename}
          \getvariable{PocketDiaryAddress}{Familyname} \NC\NR
      \NC \translate[en=Typography,
                      nl=Typografie,
                      de=Typographie,
                      fr=Typographie,
                      it=Tipografia,
                      es=Tipografía]
      \EQ \CONTEXT\ with \LUAMETATEX \NC\NR
      \NC \translate[en=Compilation date,
                      nl=Compilatie datum,
                      de=Kompilationsdatum,
                      fr=Date de compilation,
                      it=Dato di compilazione,
                      es=Fecha de compilación]
      \EQ \currentdate \NC\NR
    \stoptabulate
  \stopstandardmakeup

  \getbuffer[Lost-Returnto]
```

```
  \getbuffer[Yearcurrentplan]

  \setupheadertexts[\bfa \getvariable{PocketDiary}{Yearnext}+1]

  \getbuffer[Yearnextplan]

  \setupheadertexts[]

  \getbuffer[Yearofweektables]

  \dorecurse
    {13}
    {\getbuffer[Lines]}
\stoptext
```

After compiling, the pages look as follows:



Even Page Odd Page

Calendar with Weektables one Week on Two Facing Pages

7. Thoughts

The article has become quite lengthy. One one side it is because of the fact that all examples need quite similar setups, wich results in redundancy. The lua-code on the other side also looks comparable in all examples, at least as far as structure is concerned. Due to the fact that all examples are made up as ConTEXt-lua-documents these code-sections become quite long. Anyway I hope that I have shown that with imagination a lot is possible with the PocketDiary module.

The collection of examples will be uploaded as a module to the ConTEXt-garden.

8. Thanks

I would like to express my gratitude to Hans Hagen, who patiently helped me solve all the hick-ups while preparing these calendars. Also I would like to thank Pablo Rodriguez for providing the Spanish interface for the PocketDiary module.