

---

# **khal Documentation**

***Release 0.8.1***

**Christan Geier et al.**

May 03, 2016



<b>1</b>	<b>Features</b>	<b>3</b>
<b>2</b>	<b>Table of Contents</b>	<b>5</b>



*Khal* is a standards based CLI (console) calendar program, able to synchronize with [CalDAV](#) servers through [vdirsyncer](#).



---

## Features

---

(or rather: limitations)

- khal can read and write events/icalendars to `vdir`, so `vdirsynchroner` can be used to synchronize calendars with a variety of other programs, for example CalDAV servers.
- fast and easy way to add new events
- ikhal (interactive khal) lets you browse and edit calendars and events
- only rudimentary support for creating and editing recursion rules
- you cannot edit the timezones of events
- works with python 3.3+
- khal should run on all major operating systems <sup>1</sup>

---

<sup>1</sup> except for Microsoft Windows





---

## Table of Contents

---

### 2.1 Installing

If *khal* is packaged for your OS/distribution, using your system's standard package manager is probably the easiest way to install *khal*:

- [pkgsrc](#)
- Fedora: - Fedora 22 and later:

```
sudo dnf install -y khal
```

- Personal repos for openSUSE:

- openSUSE 13.1:

```
sudo zypper ar -f http://download.opensuse.org/repositories/home:/seilerphilipp/openSUSE_13.
```

- openSUSE 13.2:

```
sudo zypper ar -f http://download.opensuse.org/repositories/home:/seilerphilipp/openSUSE_13.
```

- AUR packages for ArchLinux: [stable](#) and [development](#) version.

If a package isn't available (or it is outdated) you need to fall back to one of the methods mentioned below.

#### 2.1.1 Install via Python's Package Managers

Since *khal* is written in python, you can use one of the package managers available to install python packages, e.g. *pip*.

You can install the latest released version of *khal* by executing:

```
pip install khal
```

or the latest development version by executing:

```
pip install git+git://github.com/pimutils/khal.git
```

This should also take care of installing all required dependencies.

## 2.1.2 Requirements

*khal* is written in python and can run on Python 3.3+. It requires a Python with `sqlite3` support enabled (which is usually the case).

If you are installing python via *pip* or from source, be aware that since *khal* indirectly depends on *lxml* you need to either install it via your system's package manager or have python's `libxml2`'s and `libxslt1`'s headers (included in a separate "development package" on some distributions) installed.

## 2.2 Configuring

**khal** reads configuration files in the *ini* syntax, meaning it understands keys separated from values by a `=`, while section and subsection names are enclosed by single or double square brackets (like `[sectionname]` and `[[subsection-name]]`).

### 2.2.1 Location of configuration file

**khal** is looking for a configuration file named *khal.conf* in the following places: in `$XDG_CONFIG_HOME/khal/` (on most systems this is `~/ .config/khal/` by default), `~/ .khal/` and in the current directory. Alternatively you can specify with configuration file to use with `-c path/to/config` at runtime.

#### The [calendars] section

The *[calendars]* is mandatory and must contain at least one subsection. Every subsection must have a unique name (enclosed by two square brackets). Each subsection needs exactly one *path* setting, everything else is optional. Here is a small example:

```
[calendars]

[[home]]
    path = ~/.calendars/home/
    color = dark green

[[work]]
    path = ~/.calendars/work/
    readonly = True
```

#### type

Set the type of this collection, the default is `calendar`. If set to `birthdays` *khal* will expect a VCARD collection and extract birthdays from those VCARDS. `birthdays` also implies `readonly=True`. If set to `calendar` only files with the `.ics` extension will be used and if set to `birthdays` only files with the `.vcf` extension will be used.

**type** option, allowed values are *calendar* and *birthdays*

**default** `calendar`

#### path

The path to an existing directory where this calendar is saved as a *vdir*. The directory is searched for events or birthdays (see `type`) but the search is not recursive.

**type** string

**default** `None`

**readonly**

setting this to *True*, will keep khal from making any changes to this calendar

**type** boolean

**default** False

**color**

khal will use this color for coloring this calendar's event. The following color names are supported: *black, white, brown, yellow, dark gray, dark green, dark blue, light gray, light green, light blue, dark magenta, dark cyan, dark red, light magenta, light cyan, light red* Depending on your terminal emulator's settings, they might look different than what their name implies. In addition to the 16 named colors an index from the 256-color palte or a 24-bit color code can be used, if your terminal supports this. The 256-color palte index is simply a number between 0 and 255. The 24-bit color must be given as #RRGGBB, where RR, GG, BB is the hexadecimal value of the red, green and blue component, respectively. When using a 24-bit color, make sure to enclose the color value in ' or '!' If the color is set to '' (the default), the default\_color (see below) is used.

**type** color

**default**

**The [sqlite] section****path**

khal stores its internal caching database here, by default this will be in the *\$XDG\_DATA\_HOME/khal/khal.db* (this will most likely be *~/.local/share/khal/khal.db*).

**type** string

**default** None

**The [locale] section**

The most important options in the the **[locale]** section are probably (long-)time and dateformat.

**unicode\_symbols**

by default khal uses some unicode symbols (as in 'non-ascii') as indicators for things like repeating events, if your font, encoding etc. does not support those symbols, set this to *False* (this will enable ascii based replacements).

**type** boolean

**default** True

**longdatetimetypeformat**

khal will display and understand all datetimes in this format, it should contain a year (e.g. *%Y*) see [timeformat](#) for the format.

**type** string

**default** %d.%m.%Y %H:%M

**weeknumbers**

Enable weeknumbers in *calendar* and *interactive* (ikhal) mode. As those are iso weeknumbers, they only work properly if *firstweekday* is set to 0

**type** weeknumbers

**default** off

#### **datetimeformat**

khal will display and understand all datetimes in this format, see [timeformat](#) for the format.

**type** string

**default** %d.%m. %H:%M

#### **dateformat**

khal will display and understand all dates in this format, see [timeformat](#) for the format

**type** string

**default** %d.%m.

#### **default\_timezone**

this timezone will be used for new events (when no timezone is specified) and when khal does not understand the timezone specified in the icalendar file. If no timezone is set, the timezone your computer is set to will be used.

**type** timezone

**default** None

#### **local\_timezone**

khal will show all times in this timezone If no timezone is set, the timezone your computer is set to will be used.

**type** timezone

**default** None

#### **longdateformat**

khal will display and understand all dates in this format, it should contain a year (e.g. %Y) see [timeformat](#) for the format.

**type** string

**default** %d.%m.%Y

#### **timeformat**

khal will display and understand all times in this format.

The formatting string is interpreted as defined by Python's [strftime](#), which is similar to the format specified in `man strftime`.

**type** string

**default** %H:%M

#### **encoding**

set this to the encoding of your terminal emulator

**type** string

**default** utf-8

#### **firstweekday**

the day first day of the week, were Monday is 0 and Sunday is 6

**type** integer, allowed values are between 0 and 6

**default** 0

## The [keybindings] section

keybindings for **ikh** are set here. You can bind more than one key (combination) to a command by supplying a comma-separated list of keys. For binding key combinations just add concatenate them (with a space in between), e.g. **ctrl n**.

### **right**

move the cursor right (in the calendar browser)

**type** list

**default** right, l, space

### **search**

open a text field to start a search for events

**type** list

**default** /

### **mark**

go into highlight (visual) mode to choose a date range

**type** list

**default** v

### **up**

move the cursor up (in the calendar browser)

**type** list

**default** up, k

### **duplicate**

duplicate the currently selected event

**type** list

**default** p

### **other**

in highlight mode go to the other end of the highlighted date range

**type** list

**default** o

### **export**

export event as ICS

**type** list

**default** e

### **new**

create a new event on the selected date

**type** list

**default** n

### **view**

show details or edit (if details are already shown) the currently selected event

**type** list

**default** enter, tab

**today**

focus the calendar browser on today

**type** list

**default** t

**left**

move the cursor left (in the calendar browser)

**type** list

**default** left, h, backspace

**delete**

delete the currently selected event

**type** list

**default** d

**down**

move the cursor down (in the calendar browser)

**type** list

**default** down, j

**save**

save the currently edited event and leave the event editor

**type** list

**default** meta enter

## The [default] section

The default section begins with a **[default]** tag. Some default values and behaviours are set here.

**default\_command**

command to be executed if no command is given when executing khal

**type** option, allowed values are *calendar*, *agenda*, *interactive*, *printformats*, *printcalendars* and *\*\**

**default** calendar

**show\_all\_days**

By default, khal displays only dates with event in “agenda” view. Setting this to *True* will show all days in “agenda”, even when there is no event

**type** boolean

**default** False

**default\_calendar**

The calendar to use if none is specified for some operation (e.g. if adding a new event). If this is not set, such operations require an explicit value.

**type** string

**default** None

**highlight\_event\_days**

If true, khal will highlight days with events. Options for highlighting are in [highlight\_days] section.

**type** boolean

**default** False

**days**

By default, khal show events for today and tomorrow. Setting this to a different value will show events of that amount of days by default.

**type** integer

**default** 2

**print\_new**

After adding a new event, what should be printed to standard out? The whole event in text form, the path to where the event is now saved or nothing?

**type** option, allowed values are *event*, *path* and *False*

**default** False

## The [view] section

The view section contains config options that effect the visual appearance when using ikhal

**event\_view\_weighting**

This is the weighting that is applied to the event view window

**type** integer

**default** 1

**frame**

Whether to show a visible frame (with *box drawing* characters) around some (groups of) elements.

**type** boolean

**default** False

**event\_view\_always\_visible**

Set to true to always show the event view window when looking at the event list

**type** boolean

**default** False

**theme**

Choose a color theme for khal.

This is very much work in progress. Help is really welcome! The two currently available color schemes (*dark* and *light*) are defined in *khal/ui/themes.py*, you can either help improve those or create a new one (see below). As ikhal uses urwid, have a look at [urwid's documentation](#) for how to set colors and/or at the existing schemes. If you cannot change the color of an element (or have any other problems) please open an issue on [github](#).

If you want to create your own color scheme, just copy the structure of the existing ones, give it a new and unique name and also add it as an option in *khal/settings/khal.spec* in the section *[default]* of the property *theme*.

**type** option, allowed values are *dark* and *light*

**default** dark

**bold\_for\_light\_color**

Whether to use bold text for light colors or not. Non-bold light colors may not work on all terminals but allow using light background colors.

**type** boolean

**default** True

**The [highlight\_days] section**

When highlight\_event\_days is enabled, this section specifies how is the highlighting rendered.

**multiple**

How to color days with events from multiple calendars - either explicit color or use calendars' colors when set to ""

**type** color

**default**

**method**

Highlighting method to use - foreground or background

**type** option, allowed values are *foreground*, *fg*, *background* and *bg*

**default** fg

**color**

What color to use when highlighting - explicit color or use calendar color when set to ""

**type** color

**default**

**default\_color**

Default color for calendars without color - when se to "" it actually disables highlighting for events that should use the default color.

**type** color

**default**

A minimal sample configuration could look like this:

**2.2.2 Example**

```
[calendars]
[[home]]
path = ~/.calendars/home/

[[work]]
path = ~/.calendars/work/

[locale]
local_timezone= Europe/Berlin
default_timezone= Europe/Berlin
timeformat= %H:%M
dateformat= %d.%m.
longdateformat= %d.%m.%Y
datetimeformat= %d.%m. %H:%M
longdatetimeformat= %d.%m.%Y %H:%M
```



### 2.2.3 syncing

To get **khal** working with CalDAV you will first need to setup **vdirsyncer**. After each start **khal** will automatically check if anything has changed and automatically update its caching db (this may take some time after the initial sync, especially for large calendar collections). Therefore you might want to execute **khal** automatically after syncing with **vdirsyncer** (e.g. via **cron**).

## 2.3 Usage

Khal offers a set of commands, most importantly **agenda**, **calendar**, **interactive**, **new**, **printcalendars**, **printformats**, and **search**. See below for a description of what every command does. Calling **khal** without any command will invoke the default command, which can be specified in the configuration file.

### 2.3.1 Options

**khal** (without any commands) has some options to print some information about **khal**:

**--version**

Prints khal's version number and exits

**-h, --help**

Prints a summary of khal's options and commands and then exits

Several options are common to almost all of **khal**'s commands (exceptions are described below):

**-v**

Be more verbose (e.g. print debugging information)

**-c** CONFIGFILE

Use an alternate configuration file

**-a** CALENDAR

Specify a calendar to use (which must be configured in the configuration file), can be used several times. Calendars not specified will be disregarded for this run.

**-d** CALENDAR

Specify a calendar which will be disregarded for this run, can be used several times.

**--color/-no-color**

**khal** will detect if standard output is not a tty, e.g., you redirect khal's output into a file, and if so remove all highlighting/coloring from its output. Use **--color** if you want to force highlighting/coloring and **--no-color** if you want coloring always removed.

### 2.3.2 dates

Almost everywhere khal accepts dates, khal should recognize relative date names like *today*, *tomorrow* and the names of the days of the week (also in three letters abbreviated form). Week day names get interpreted as the date of the next occurrence of a day with that name. The name of the current day gets interpreted as that date *next* week (i.e. seven days from now).

### 2.3.3 Commands

#### agenda

shows all events scheduled for given dates. `khal agenda` should understand the following syntax:

```
khal agenda [-a CALENDAR ... | -d CALENDAR ...] [--days N] [DATE ...]
```

If no dates are supplied as arguments, today and tomorrow are used. Dates must be given in the format specified in khal's config file as *dateformat* or *longdateformat*. If *dateformat* is used, the current year is implied.

**--days** N

Specify how many days' (following each DATE) events should be shown.

#### at

shows all events scheduled for a given datetime. `khal at` should be supplied with a date and time, a time (the date is then assumed to be today) or the string *now*. `at` defaults to *now*.

```
khal at [-a CALENDAR ... | -d CALENDAR ...] [DATETIME | now]
```

#### calendar

shows a calendar (similar to `cal(1)`) and agenda. `khal calendar` should understand the following syntax:

```
khal calendar [-a CALENDAR ... | -d CALENDAR ...] [--days N] [DATE ...]
```

Date selection works exactly as for `khal agenda`. The displayed calendar contains three consecutive months, where the first month is the month containing the first given date. If today is included, it is highlighted. Have a look at `khal agenda` for a description of the options.

#### configure

will help users creating an initial configuration file. **configure** will refuse to run if there already is a configuration file.

#### import

lets the user import `.ics` files with the following syntax:

```
khal import [-a CALENDAR] [--batch] [--random-uid|-r] ICSFILE
```

If an event with the same UID is already present in the (implicitly) selected calendar `khal import` will ask before updating (i.e. overwriting) that old event with the imported one, unless `--batch` is given, then it will always update. If this behaviour is not desired, use the `--random-uid` flag to generate a new, random UID. If no calendar is specified (and not `--batch`), you will be asked to choose a calendar. You can either enter the number printed behind each calendar's name or any unique prefix of a calendar's name.

## interactive

invokes the interactive version of khal, can also be invoked by calling **ikhal**. While ikhal can be used entirely with the keyboard, some elements respond if clicked on with a mouse (mostly by being selected).

When the calendar on the left is in focus, you can

- move through the calendar (default keybindings are the arrow keys, space and backspace, those keybindings are configurable in the config file)
- focus on the right column by pressing tab or enter
- re-focus on the current date, default keybinding t as in today
- marking a date range, default keybinding v, as in visual, think visual mode in Vim, pressing esc escape this visual mode
- if in visual mode, you can select the other end of the currently marked range, default keybinding o as in other (again as in Vim)
- create a new event on the currently focused day (or date range if a range is selected), default keybinding n as in new
- search for events, default keybinding /, a pop-up will ask for your search term

When an event list is in focus, you can

- view an event's details with pressing enter (or tab) and edit it with pressing enter (or tab) again (if [default] event\_view\_always\_visible is set to True, the event in focus will always be shown in detail)
- toggle an event's deletion status, default keybinding d as in delete, events marked for deletion will appear with a D in front and will be deleted when khal exits.
- duplicate the selected event, default keybinding p as in duplicate (d was already taken)
- export the selected event, default keybinding e

In the event editor, you can

- jump to the next (previous) selectable element with pressing tab (shift+tab)
- quick save, default keybinding meta+enter (meta will probably be alt)
- use some common editing short cuts in most text fields (ctrl+w deletes word before cursor, ctrl+u (ctrl+k) deletes till the beginning (end) of the line, ctrl+a (ctrl+e) will jump to the beginning (end) of the line
- in the date and time field you can increment and decrement the number under the cursor with ctrl+a and ctrl+x (time in 15 minute steps)
- activate actions by pressing enter on text enclosed by angled brackets, e.g. < Save > (sometimes this might open a pop up)

Pressing esc will cancel the current action and/or take you back to the previously shown pane (i.e. what you see when you open ikhal), if you are at the start pane, ikhal will quit on pressing esc again.

## new

allows for adding new events. khal new should understand the following syntax:

```
khal new [-a CALENDAR] [OPTIONS] startdatetime [enddatetime] [timezone] summary [description]
```

where start- and enddatetime are either datetimes, times, or keywords and times in the formats defined in the config file. If no calendar is given via `-a`, the default calendar is used. **new** does not support `-d` and also `-a` may only be used once.

**new** accepts these combinations for start and endtimes (specifying the end is always optional):

- *datetime [datetime|time] [timezone]*
- *time [time] [timezone]*
- *date [date]*

where the formats for datetime and time are as follows:

- *datetime = (longdatetimeformat\datetimeformat|keyword-date timeformat)*
- *time = timeformat*
- *date = (longdateformat\dateformat)*

and *timezone*, which describes the timezone the events start and end time are in, should be a valid Olson DB identifier (like *Europe/Berlin* or *America/New\_York*. If no timezone is given, the *defaulttimezone* as configured in the configuration file is used instead.

The exact format of longdatetimeformat, datetimeformat, timeformat, longdateformat and dateformat can be configured in the configuration file. Valid keywords for dates are *today*, *tomorrow*, the English name of all seven weekdays and their three letter abbreviations (their next occurrence is used).

If no end is given, the default length of one hour or one day (for all-day events) is used. If only a start time is given the new event is assumed to be starting today. If only a time is given for the event to end on, the event ends on the same day it starts on, unless that would make the event end before it has started, then the next day is used as end date

If a 24:00 time is configured (timeformat = %H:%M) an end time of 24:00 is accepted as the end of a given date.

If the **summary** contains the string `::`, everything after `::` is taken as the **description** of the new event, i.e., the “body” of the event (and `::` will be removed).

## Options

- **-l, -location=LOCATION** specify where this event will be held.
- **-r, -repeat=RRULE** specify if and how this event should be recurring. Valid values for *RRULE* are *daily*, *weekly*, *monthly* and *yearly*
- **-u, -until=UNTIL** specify until when a recurring event should run
- **-alarm DURATION** will add an alarm DURATION before the start of the event, *DURATION* should look like *1day 10minutes* or *1d3H10m*, negative *DURATIONs* will set alarm after the start of the event.

## Examples

```
khal new 18:00 Awesome Event
```

adds a new event starting today at 18:00 with summary ‘awesome event’ (lasting for the default time of one hour) to the default calendar

```
khal new tomorrow 16:30 Coffee Break
```

adds a new event tomorrow at 16:30

```
khal new 25.10. 18:00 24:00 Another Event :: with Alice and Bob
```

adds a new event on 25th of October lasting from 18:00 to 24:00 with an additional description

```
khal new -a work 26.07. Great Event -r weekly
```

adds a new all day event on 26th of July to the calendar *work* which recurs every week.

### printcalendars

prints a list of all configured calendars.

### printformats

prints a fixed date (*2013-12-11 10:09*) in all configured date(time) formats. This is supposed to help check if those formats are configured as intended.

### search

search for events matching a search string and print them. Currently recurring events are only printed once. No advanced search features are currently supported.

The command

```
khal search party
```

prints all events matching *party*.

## 2.4 Standard Compliance

*khal* tries to follow standards and RFCs (most importantly [RFC 5545](#) *iCalendar*) where ever possible. Known intentional and unintentional deviations are listed below.

### 2.4.1 RDATE;VALUE=PERIOD

*RDATE* s with *PERIOD* values are currently not supported, as **icalendar\_** does does not support it yet. Please submit any real world examples of events with *RDATE;VALUE=PERIOD* you might encounter (*khal* will print warnings if you have any in your calendars).

### 2.4.2 RANGE=THISANDPRIOR

Recurrent events with the *RANGE=THISANDPRIOR* are and will not be <sup>1</sup> supported by *khal*, as applications supporting the latest [standard](#) MUST NOT create those. *khal* will print a warning if it encounters an event containing *RANGE=THISANDPRIOR*.

---

<sup>1</sup> unless a lot of users request this feature

### 2.4.3 Events with neither END nor DURATION

While the RFC states:

A calendar entry with a "DTSTART" property but no "DTEND" property does not take up any time. It is intended to represent an event that is associated with a given calendar date and time of day, such as an anniversary. Since the event does not take up any time, it MUST NOT be used to record busy time no matter what the value for the "TRANSP" property.

khal transforms those events into all-day events lasting for one day (the start date). As long as those events do not get edited, these changes will not be written to the vdir (and with that to the CalDAV server). Any timezone information that was associated with the start date gets discarded.

---

**Note:** While the main rationale for this behaviour was laziness on part of the main author of khal, other calendar software shows the same behaviour (e.g. Google Calendar and Evolution).

---

## 2.5 Timezones

Getting localized time right, seems to be the most difficult part about calendaring (and messing it up ends in missing the one important meeting of the week). So I'll briefly describe here, how khal tries to handle timezone information, which information it can handle and which it can't.

In general, there are two different type of events. *Localized events* (with *localized* start and end datetimes) which have timezone information attached to their start and end datetimes, and *floating events* (with *floating* start and end datetimes), which have no timezone information attached (all-day events, events that last for complete days are floating as well). Localized events are always observed at the same **UTC** (no matter what time zone the observer is in), but different local times. On the other hand, floating events are always observed at the same local time, which might be different in UTC.

In khal all localized datetimes are saved to the local database as UTC. Datetimes that are already UTC, e.g. 19980119T070000Z, are saved as such, others are converted to UTC (but don't worry, the timezone information does not get lost). Floating events get saved in floating time, independently of the localized events.

If you want to look up which events take place at a specified datetime, khal always expects that you want to know what events take place at that *local* datetime. Therefore, the datetime you asked for gets converted to UTC, the appropriate *localized* events get selected and presented with their start and end datetimes *converted* to *your local datetime*. For floating events no conversion is necessary.

Khal (i.e. **icalendar**\_) can understand all timezone identifiers as used in the **Olson DB** and custom timezone definitions, if those VTIMEZONE components are placed before the VEVENTS that make use of them (as most calendar programs seem to do). In case a unknown (or unsupported) timezone is found, khal will assume you want that event to be placed in the *default timezone* (which can be configured in the configuration file as well).

khal expects you *always* want *all* start and end datetimes displayed in *local time* (which can be set in the configuration file as well, otherwise your computer's timezone is used).

## 2.6 Feedback and Contributing

The preferred way of interaction with *khal*'s developers is via [github](#). But if you do not feel comfortable using github, please don't hesitate to contact the original developer via email (khal at lostpackets dot de) or via jabber/XMPP (geier at jabber dot ccc dot de).

### 2.6.1 Feedback

Please do provide feedback if *khal* works for you or even more importantly if it doesn't. Feature requests and other ideas on how to improve *khal* are also welcome.

In case you are not satisfied with *khal*, there are at least two other projects with similar aim you might want to check out: [calendar-cli](#) (no offline storage and a bit different scope) and [gcalcli](#) (only works with google's calendar).

### 2.6.2 Submitting a Bug

If you found a bug or any part of *khal* isn't working as you expected, please check if that bug is either already reported at

<https://github.com/pimutils/khal/issues?state=open>

If it isn't, always run the command again with the `-v` option and paste the output of that (of course you can edit out any private details).

### 2.6.3 Hacking

If you want to hack on *khal* and have some questions, please don't hesitate to contact us, we will gladly provide you with any information you need.

Before submitting your first patch, please add yourself to *AUTHORS.txt*.

You can submit patches either via github pull requests (the preferred method) or via email (*khal* at lostpackets dot de).

## 2.7 Notes for developers

If you want to hack on *khal*, the notes below might be of some interest to you.

### 2.7.1 iCalendar peculiarities

A single *.ics* can contain several VEVENTS, which might or might not be the part of the same event. This can lead to issues with straight forward implementations. Some of these, and the way *khal* is dealing with them, are described below.

While one would expect every VEVENT to have its own unique UID (for what it's worth they are named *unique identifier*), there is a case where several VEVENTS have the same UID, but do describe the same (recurring) event. In this case, one VEVENT, containing an RRULE or RDATE element would be the *proto* event, from which all recurrence instances are derived. All other VEVENTS with the same UID would then have a RECURRENCE-ID element (I'll call them *child* event from now on) and describe deviations of at least one recurrence instance (RECURRENCE-ID elements can also have the added property RANGE=THISANDFUTURE, meaning the deviations described by this child event also apply to all further recurrence instances).

Because it is possible that an event already in the database consists of a master event and at least one child event gets updated and then consists only of a master event, we currently *delete* all events with the same UID from the database when inserting or updating a new event. But this means that we need to update an event always at once (master and all child events) at the same time (using *Calendar.update()* or *Calendar.new()* in this case)

As this wouldn't be bad enough, the standard loses no words on the ordering on those VEVENTS in any given *.ics* file (at least I didn't find any). Not only can the proto event be *behind* any or all RECURRENCE-ID events, but also events with different UIDs can be in between.

We therefore currently first collect all events with the same UID and than sort those by their type (proto or child), and the children by the value of the RECURRENCE-ID property.

## 2.8 Changelog

All notable changes to this project should be documented here. For more detailed information have a look at the git log.

Package maintainers and users who have to manually update their installation may want to subscribe to [GitHub's tag feed](#).

### 2.8.1 0.8.1

released on 2015-04-13

- fix bug in `CalendarWidget.set_focus_date()` (Christian Geier)

### 2.8.2 0.8.0

released on 2015-04-13

- BREAKING CHANGE: python 2 is no longer supported (Hugo Osvaldo Barrera)
- updated dependency: `vdirsynchroner`  $\geq$  0.5.2
- make tests work with `icalendar` 3.9.2 (no functional changes) (Christian Geier)
- new dependency: `freezegun` (only for running the tests)
- khal's git repository moved to <https://github.com/pimutils/khal>
- support for showing the birthday of contacts with no FN property (Hugo Osvaldo Barrera)
- increased start up time when coloring is enabled (Christian Geier)
- improved color support (256 colors and 24-bit colors), see configuration documentation for details (Sebastian Hamann)
- renamed color *grey* to *gray* (Sebastian Hamann)
- in *khal new* treat 24:00 as the end of a day/00:00 of the next (Christian Geier)
- new allowed value for a calendar's color: *auto* (also the new default), if set, khal will try to read a file called *color* from that calendar's vdir (see `vdirsynchroner`'s documentation on *metasync*). If that file is not present or its contents is not understood, the default color will be used (Christian Geier)
- new allowed value for calendar's type: *discover*, if set, khal will (recursively) search that calendar's path for valid vdirs and add those to the configured calendars (Christian Geier)
- new command *configure* which should help new users set up a configuration file (Christian Geier)
- warn user when parsing broken icalendar files, this requires `icalendar`  $>$  3.9.2 (Christian Geier)
- khal will now strip all ANSI escape codes when it detects that stdout is no tty, this behaviour can be overwritten with the new options `-color/ -no-color` (Markus Unterwaditzer)
- calendar and agenda have a new option `-week`, if set all events from current week (or the week containing the given date) are shown (Stephan Weller)
- new option `-alarm DURATION` for *new* (Max Voit)



## ikhal

- basic export of events from event editor pane and from event lists (default keybinding: *e* (Filip Pytloun))
- pressing *enter* in a date editing widget will now open a small calendar widget, arrow keys can be used to select a date, enter (or escape) will close it again (Christian Geier)
- in highlight/date range selection mode the other end can be selected, default keybinding *o* (as in *Other*) (Christian Geier)
- basic search is now supported (default keybinding */*) (Christian Geier)
- in the event editor and pop-up Dialogs select the next (previous) item with tab (shift tab) (Christian Geier)
- only allow saving when starttime < endtime (Christian Geier)
- the event editor now allows editing of alarms (but khal will not actually alarm you at the given time) (Johannes Wienke)

## 2.8.3 0.7.0

released on 2015-11-24

There are no new or dropped dependencies.

- most of the internal representation of events was rewritten, the current benefit is that floating events are properly represented now, hopefully more is to come (Christian Geier)
- *printformats* uses a more sensible date now (John Shea)
- khal and ikhal can now highlight dates with events, at the moment, enabling it does noticeably slow down (i)khal's start; set `[default] highlight_event_days = True` and see section `[highlight_days]` for further configuration (Dominik Joe Pantůček)
- fixed line wrapping for *at* (Thomas Schape)
- *calendar* and *agenda* optionally print location and description of all events, enable with the new `-full/-f` flag (Thomas Schaper)
- updated and improved zsh completion file (Oliver Kiddle)
- FIX: deleting events did not always work if an event with the same filename existed in an other calendar (but no data lost incurred) (Christian Geier)

## ikhal

- events are now displayed nicer (Thomas Glanzmann)
- support for colorschemes, a *light* and *dark* one are currently included, help is wanted to make them prettier and more functional (config option `[view] theme: (dark|light)`) (Christian Geier)
- ikhal can now display frames around some user interface elements, making it nicer to look at in some eyes (config option `[view] frame: True`) (Christian Geier)
- events can now be duplicated (default keybinding: *p*) (Christian Geier)
- events created while time ranges are selected (default keybinding to enable date range selection: *v*) will default to that date range (Christian Geier)
- when trying to delete recurring events, users are now asked if they want to delete the complete event or just this instance (Christian Geier)

## 2.8.4 0.6.0

2015-07-15

- BUGFIX Recurrent events with a THISANDFUTURE parameter could affect other events. This could lead to events not being found by the normal lookup functionality when they should and being found when they shouldn't. As the second case should result in an error that nobody reported yet, I hope nobody got bitten by this.
- new dependency for running the tests: freezegun
- new dependency for setup from scm: setuptools\_scm
- khal now needs to be installed for building the documentation
- ikhal's should now support ctrl-e, ctrl-a, ctrl-k and ctrl-u in editable text fields (Thomas Glanzmann)
- ikhal: space and backspace are new (additional) default keybindings for right and left (Pierre David)
- when editing descriptions you can now insert new lines (Thomas Glanzmann)
- khal should not choose an arbitrary default calendar anymore (Markus Unterwaditzer)
- the zsh completion file has been updated (Hugo Osvaldo Barrera)
- new command *import* lets users import .ics files (Christian Geier)
- khal should accept relative dates on the command line (today, tomorrow and weekday names) (Christian Geier)
- keybinding for saving an event from ikhal's event editor (default is *meta + enter*) (Christian Geier)

## 2.8.5 0.5.0

released on 2015-06-01

- fixed several bugs relating to events with unknown timezones but UNTIL, RDATE or EXDATE properties that are in Zulu time (thanks to Michele Baldessari for reporting those)
- bugfix: on systems with a local time of UTC-X dealing with allday events lead to crashes
- bugfix: British summer time is recognized as daylight saving time (Bradley Jones)
- compatibility with vdirsncr 0.5
- new command *search* allows to search for events
- user changeable keybindings in ikhal, with hjkl as default alternatives for arrows in calendar browser, see documentation for more details
- new command *at* shows all events scheduled for a specific datetime
- support for reading birthdays from vcard collections (set calendar/collection *type* to *birthdays*)
- new command *printformats* prints a fixed date in all configured date-time settings
- *new* now supports the *-until-u* flag to specify until when recurring events should run (Micah Nordland)
- python 3 (>= 3.3) support (Hugo Osvaldo Barrera)

### ikhal

- minimal support for recurring events in ikhal's editor (Micah Nordland)
- configurable view size in ikhal (Bradley Jones)

- show events organizers (Bradley Jones)
- major reorganisation of ikhal layout (Markus Unterwaditzer)

## 2.8.6 0.4.0

released on 2015-02-02

### dependency changes

- new dependency: click>3.2
- removed dependency: docopt
- note to package maintainers: *requirements.txt* has been removed, dependencies are still listed in *setup.py*

### note to users

- users will need to delete the local database, no data should be lost (and khal will inform the user about this)

### new and changed features

- new config\_option: *[default] print\_new*, lets the user decide what should be printed after adding a new event
- new config option: *[default] show\_all\_days* lets users decide if they want to see days without any events in agenda and calendar view (thanks to Pierre David)
- khal (and ikhal) can now display weeknumbers
- khal new can now create repetitive events (with `-repeat`), see documentation (thanks to Eric Scheibler)
- config file: the debug option has been removed (use *khal -v* instead)
- FIX: vttimezones were not assembled properly, this lead to spurious offsets of events in some other calendar applications
- change in behaviour: recurring events are now always expanded until 2037
- major speedup in inserting events into the caching database, especially noticeable when running khal for the first time or after an deleting the database (Thanks to Markus Unterwaditzer)
- better support for broken events, e.g. events ending before they start (Thanks to Markus Unterwaditzer)
- more recurrence rules are supported, khal will print warnings on unsupported rules

### ikhal

- ikhal's calendar should now be filled on startup
- pressing *t* refocuses on today
- pressing `ctrl-w` in input fields should delete the last word before the cursor
- when the focus is set on the events list/editor, the current date should still be visible in the calendar

## 2.8.7 0.3.1

released on 2014-09-08

- **FIX:** events deleted in the vdir are not shown anymore in khal. You might want to delete your local database file, if you have deleted any events on the server.
- **FIX:** in some cases non-ascii characters were printed even if `unicode_symbols` is set to `False` in the config
- **FIX:** events with different start and end timezones are now properly exported (the end timezone was disregarded when building an icalendar, but since timezones cannot be edited anyway, this shouldn't have caused any problems)
- **FIX:** calendars marked as read-only in the configuration file should now really be read-only

## 2.8.8 0.3.0

released on 2014-09-03

- **new unified documentation**
  - html documentation (website) and man pages are all generated from the same sources via sphinx (type *make html* or *make man* in *doc/*, the result will be build in *build/html* or *build/man* respectively
  - the new documentation lives in *doc/*
  - the package *sphinxcontrib-newsfeed* is needed for generating the html version (for generating an RSS feed)
  - the man pages live *doc/build/man/*, they can be build by running *make man* in *doc/sphinx/*
- new dependencies: *configobj*, *tzlocal* >=1.0
- **IMPORTANT:** the configuration file's syntax changed (again), have a look at the new documentation for details
- *local\_timezone* and *default\_timezone* will now be set to the timezone the computer is set to (if they are not set in the configuration file)

## 2.9 Frequently Asked Questions (FAQ)

- **start up of khal and ikhal is very slow** In some case the *pytz* (python timezone) is only available as a zip file, as *pytz* accesses several parts during initialization this takes some time. If *time python -c "import pytz; pytz.timezone('Europe/Berlin')"* takes nearly as much time as running *khal*, uncompressing that file via *pytz* via *(sudo) pip unzip pytz* might help.
- **ikhal raises an Exception: AttributeError: 'module' object has no attribute 'SimpleFocusListWalker'**  
You probably need to upgrade *urwid* to version 1.1.0, if your OS does come with an older version of *urwid* you can install the latest version to userspace (with out messing up your default installation) with *pip install --upgrade urwid --user*.
- **Installation stops with an error: source/str\_util.c:25:20: fatal error: Python.h: No such file or directory**  
You do not have the Python development headers installed, on Debian based Distributions you can install them via *aptitude install python-dev*.

## 2.10 License

khal is released under the Expat/MIT License:

```
Copyright (c) 2013-2015 Christian Geier et al.
```

```
Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:
```

```
The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.
```

```
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
```

## 2.11 News

Below is a list of new releases and other khal related news. This is also available as an [rss feed](#) .

### 2.11.1 khal v0.8.1 released

Published on *2016-04-13*

The second version released today ([khal v0.8.1](#), yes, also on [pypi](#)) fixes a bug in the `CalendarWidget()` that probably would not have triggered but made the tests fail.

### 2.11.2 khal v0.8.0 released

Published on *2016-04-13*

The latest version of khal has been released: [khal v0.8.0](#) (as always, also on [pypi](#)).

We have recently dropped python 2 support, so this release is the first one that only supports python 3 (3.3+).

There is one more backwards incompatible change: The color *grey* has been renamed to *gray*, if you use it in your configuration file, you will need to update to *gray*.

There are some new features that should be configuring khal easier, especially for new users (e.g., new command *configure* helps with the initial configuration). Also alarms can now be entered either when creating new events with *new* or when editing them in *ikhal*.

Have a look at the changelog for more complete list of new features (of which there are many).

### 2.11.3 khal v0.7.0 released

Published on 2015-11-24

The latest version of khal has been released: [khal v0.7.0](#) (as always, also on [pypi](#)).

This release brings a lot of new features, by an ever increasing number of new contributors (welcome everyone!). With highlighting of days that have events we now have one of the most requested features implemented (because it does noticeably slow down khal's start it is disabled by default, startup performance will hopefully be increased soon). Among the other new features are duplicating events (in ikhal), prettier event display (also in ikhal) and a better zsh completion file.

Have a look at the changelog for more complete list of new features.

### 2.11.4 khal v0.6.0 released

Published on 2015-07-15

Only six weeks after the last version [khal v0.6.0](#) is now available (yes, also on [pypi](#)).

This release fixes an unfortunate bug which could lead to wrong shifts in other events when inserting complicated recurring events. All users are therefore advised to quickly upgrade to khal 0.6.

There are also quite a bunch of new features, among other nicer editing capabilities in ikhal's text edits and import of .ics files. For a more detailed list of changes, please have a look at the changelog (especially if you package khal).

### 2.11.5 khal v0.5.0 released

Published on 2015-06-01

A new release of khal is here: [khal v0.5.0](#) (also available on [pypi](#)).

This release brings a lot of new features (like rudimentary search support, user changeable keybindings in ikhal, new command *at*), python 3 support and some assorted bugfixes.

Thanks to everybody who contributed with bug reports, suggestions and code, especially to everyone contributing for the first time!

For a more detailed list of changes, please have a look at the changelog.

### 2.11.6 khal v0.4.0 released

Published on 2015-02-02

A new release of khal is here: [khal v0.4.0](#) (also available on [pypi](#)).

This release offers several functional improvements like better support for recurring events or a major speedup when creating the caching database and some new features like week number support or creating recurring events with *khal new -repeat*.

#### Note to users

khal now requires [click](#) instead of [docopt](#) and, as usual, the local database will need to be deleted.

For a more detailed list of changes, please have a look at the [Changelog](#).

## 2.11.7 pycarddav and khal at 31c3

Published on 2014-12-09

If you will be at [31C3](#) and would like to discuss the faults and merits of khal or pycarddav, commandline calendar-ing/addressbooking in general, your ideas or just have a beer or mate, I'd love to meet up. You can find my contact details under *Feedback*.

## 2.11.8 khal v0.3.1 released

Published on 2014-09-08

A new release of khal is here: [khal v0.3.1](#) (also available on [pypi](#)).

This is a bugfix release, bringing no new features. The last release suffered from a major bug, where events deleted on the server (and in the vdir) were not deleted in khal's caching database and therefore still displayed in khal. Therefore, after updating please delete your local database.

For more information on other fixed bugs, see *Changelog*.

## 2.11.9 khal v0.3 released

Published on 2014-09-03

A new release of khal is here: [khal v0.3.0](#) (also available on [pypi](#)).

If you want to update your installation from [pypi](#), you can run `sudo pip install --upgrade khal`.

## CHANGELOG

- **new unified documentation**
  - html documentation (website) and man pages are all generated from the same sources via sphinx (type `make html` or `make man` in `doc/`, the result will be build in `build/html` or `build/man` respectively (also available on [Read the Docs](#))
  - the new documentation lives in `doc/`
  - the package `sphinxcontrib-newsfeed` is needed for generating the html version (for generating an RSS feed)
  - the man pages live `doc/build/man/`, they can be build by running `make man` in `doc/sphinx/`
- new dependencies: `configobj`, `tzlocal` >=1.0
- **IMPORTANT:** the configuration file's syntax changed (again), have a look at the new documentation for details
- `local_timezone` and `default_timezone` will now be set to the timezone the computer is set to (if they are not set in the configuration file)

## 2.11.10 khal v0.2 released

Published on 2014-06-27

A new release of khal is here: [khal v0.2.0](#) (also available on [pypi](#)).

If you want to update your installation from [pypi](#), you can run `sudo pip install --upgrade khal`.

From now on *khal* relies on [vdirsyncer](#) for CalDAV sync. While this makes *khal* a bit more complicated to setup, *vdirsyncer* is much better tested than *khal* and also the [bus factor](#) increased (at least for parts of the project).

You might want to head over to the [tutorial](#) on how to setup *vdirsyncer*. Afterwards you will need to re-setup your *khal* configuration (copy the new example config file), also you will need to delete your old (local) database, so please make sure you did sync everything.

Also *khal*'s command line syntax changed quite a bit, so you might want to head over to the [documentation](#).

### 2.11.11 khal v0.1.1 released

Published on 2014-05-07

A small bugfix release: [khal v0.1.0](#) Example config file now in source dist.

### 2.11.12 khal v0.1 released

Published on 2014-04-03

The first release of *khal* is here: [khal v0.1.0](#) (and also available on [pypi](#) now).

The next release, hopefully coming rather sooner than later, will get rid of its own CalDAV implementation, but instead use *vdirsyncer*; you can already try it out via checking out the branch *vdir* at [github](#).

### 2.11.13 pycarddav and khal at 30c3

Published on 2013-12-13

If you will be [30C3](#) and would like to discuss the faults and merits of *khal* or *pycarddav*, commandline calendar-ing/addressbooking in general, your ideas or just have a beer or mate, I'd love to meet up. You can find my contact details under *Feedback*.

### 2.11.14 Call for Testing

Published on 2013-11-19

While there isn't a release yet, *khal* is, at least partly, in a usable shape by now. Please report any errors you stumble upon and improvement suggestions you have either via email or [github](#) (if you don't have any privacy concerns etc. I'd prefer you use github since it is public, but I'll soon set up a mailing list). [TODO.rst](#) gives you an idea about the plans I currently have for *khal*'s near future.

### 2.11.15 khal v0.8.1 released

Published on 2016-04-13

The second version released today ([khal v0.8.1](#), yes, also on [pypi](#)) fixes a bug in the `CalendarWidget()` that probably would not have triggered but made the tests fail.



### 2.11.16 khal v0.8.0 released

Published on 2016-04-13

The latest version of khal has been released: [khal v0.8.0](#) (as always, also on [pypi](#)).

We have recently dropped python 2 support, so this release is the first one that only supports python 3 (3.3+).

There is one more backwards incompatible change: The color *grey* has been renamed to *gray*, if you use it in your configuration file, you will need to update to *gray*.

There are some new features that should be configuring khal easier, especially for new users (e.g., new command *configure* helps with the initial configuration). Also alarms can now be entered either when creating new events with *new* or when editing them in ikhal.

Have a look at the changelog for more complete list of new features (of which there are many).

### 2.11.17 khal v0.7.0 released

Published on 2015-11-24

The latest version of khal has been released: [khal v0.7.0](#) (as always, also on [pypi](#)).

This release brings a lot of new features, by an ever increasing number of new contributors (welcome everyone!). With highlighting of days that have events we now have one of the most requested features implemented (because it does noticeably slow down khal's start it is disabled by default, startup performance will hopefully be increased soon). Among the other new features are duplicating events (in ikhal), prettier event display (also in ikhal) and a better zsh completion file.

Have a look at the changelog for more complete list of new features.

### 2.11.18 khal v0.6.0 released

Published on 2015-07-15

Only six weeks after the last version [khal v0.6.0](#) is now available (yes, also on [pypi](#)).

This release fixes an unfortunate bug which could lead to wrong shifts in other events when inserting complicated recurring events. All users are therefore advised to quickly upgrade to khal 0.6.

There are also quite a bunch of new features, among other nicer editing capabilities in ikhal's text edits and import of .ics files. For a more detailed list of changes, please have a look at the changelog (especially if you package khal).

### 2.11.19 khal v0.5.0 released

Published on 2015-06-01

A new release of khal is here: [khal v0.5.0](#) (also available on [pypi](#)).

This release brings a lot of new features (like rudimentary search support, user changeable keybindings in ikhal, new command *at*), python 3 support and some assorted bugfixes.

Thanks to everybody who contributed with bug reports, suggestions and code, especially to everyone contributing for the first time!

For a more detailed list of changes, please have a look at the changelog.

### 2.11.20 khal v0.4.0 released

Published on 2015-02-02

A new release of khal is here: [khal v0.4.0](#) (also available on [pypi](#)).

This release offers several functional improvements like better support for recurring events or a major speedup when creating the caching database and some new features like week number support or creating recurring events with *khal new -repeat*.

#### Note to users

khal now requires [click](#) instead of [docopt](#) and, as usual, the local database will need to be deleted.

For a more detailed list of changes, please have a look at the [Changelog](#).

### 2.11.21 pycarddav and khal at 31c3

Published on 2014-12-09

If you will be at [31C3](#) and would like to discuss the faults and merits of khal or pycarddav, commandline calendaring/addressbooking in general, your ideas or just have a beer or mate, I'd love to meet up. You can find my contact details under *Feedback*.

### 2.11.22 khal v0.3.1 released

Published on 2014-09-08

A new release of khal is here: [khal v0.3.1](#) (also available on [pypi](#)).

This is a bugfix release, bringing no new features. The last release suffered from a major bug, where events deleted on the server (and in the vdir) were not deleted in khal's caching database and therefore still displayed in khal. Therefore, after updating please delete your local database.

For more information on other fixed bugs, see [Changelog](#).

### 2.11.23 khal v0.3 released

Published on 2014-09-03

A new release of khal is here: [khal v0.3.0](#) (also available on [pypi](#)).

If you want to update your installation from [pypi](#), you can run *sudo pip install --upgrade khal*.

#### CHANGELOG

- **new unified documentation**

- html documentation (website) and man pages are all generated from the same sources via sphinx (type *make html* or *make man* in *doc/*, the result will be build in *build/html* or *build/man* respectively (also available on [Read the Docs](#))
- the new documentation lives in *doc/*
- the package sphinxcontrib-newsfeed is needed for generating the html version (for generating an RSS feed)

- the man pages live doc/build/man/, they can be build by running *make man* in doc/sphinx/
- new dependencies: configobj, tzlocal>=1.0
- **IMPORTANT:** the configuration file's syntax changed (again), have a look at the new documentation for details
- local\_timezone and default\_timezone will now be set to the timezone the computer is set to (if they are not set in the configuration file)

### 2.11.24 khal v0.2 released

Published on 2014-06-27

A new release of khal is here: [khal v0.2.0](#) (also available on [pypi](#)).

If you want to update your installation from [pypi](#), you can run *sudo pip install --upgrade khal*.

From now on *khal* relies on [vdirsyncer](#) for CalDAV sync. While this makes *khal* a bit more complicated to setup, *vdirsyncer* is much better tested than *khal* and also the [bus factor](#) increased (at least for parts of the project).

You might want to head over to the [tutorial](#) on how to setup *vdirsyncer*. Afterwards you will need to re-setup your *khal* configuration (copy the new example config file), also you will need to delete your old (local) database, so please make sure you did sync everything.

Also *khal*'s command line syntax changed quite a bit, so you might want to head over the [documentation](#).

### 2.11.25 khal v0.1.1 released

Published on 2014-05-07

A small bugfix release: [khal v0.1.0](#) Example config file now in source dist.

### 2.11.26 khal v0.1 released

Published on 2014-04-03

The first release of khal is here: [khal v0.1.0](#) (and also available on [pypi](#) now).

The next release, hopefully coming rather sooner than later, will get rid of its own CalDAV implementation, but instead use [vdirsyncer](#); you can already try it out via checking out the branch *vdir* at [github](#).

### 2.11.27 pycarddav and khal at 30c3

Published on 2013-12-13

If you will be [30C3](#) and would like to discuss the faults and merits of khal or pycarddav, commandline calendaring/addressbooking in general, your ideas or just have a beer or mate, I'd love to meet up. You can find my contact details under *Feedback*.

### 2.11.28 Call for Testing

Published on 2013-11-19

While there isn't a release yet, *khal* is, at least partly, in a usable shape by now. Please report any errors you stumble upon and improvement suggestions you have either via email or [github](#) (if you don't have any privacy concerns etc).

I'd prefer you use github since it is public, but I'll soon set up a mailing list). [TODO.rst](#) gives you an idea about the plans I currently have for *khal*'s near future.

## Symbols

- color/--no-color
  - command line option, 13
- days N
  - command line option, 14
- version
  - command line option, 13
- a CALENDAR
  - command line option, 13
- c CONFIGFILE
  - command line option, 13
- d CALENDAR
  - command line option, 13
- h, --help
  - command line option, 13
- v
  - command line option, 13

## C

- command line option
  - color/--no-color, 13
  - days N, 14
  - version, 13
  - a CALENDAR, 13
  - c CONFIGFILE, 13
  - d CALENDAR, 13
  - h, --help, 13
  - v, 13

## R

- RFC
  - RFC 5545, 17