NAME

calendar - reminder service

SYNOPSIS

calendar [**-A** *num*] [**-a**] [**-B** *num*] [**-D** *moon*/*sun*] [**-d**] [**-F** *friday*] [**-f** *calendarfile*] [**-l** *longitude*]
[**-t** *dd*[.*mm*[.*year*]]] [**-U** *UTC-offset*] [**-W** *num*]

DESCRIPTION

The **calendar** utility checks the current directory for a file named *calendar* and displays lines that fall into the specified date range. On the day before a weekend (normally Friday), events for the next three days are displayed.

The following options are available:

-A *num*

Print lines from today and the next *num* days (forward, future).

-a   Process the ''calendar'' files for users found in */etc/passwd* and mail the results to them. This can result in multiple messages for specific files, since */etc/passwd* does not require home directories to be unique. In particular, by default *root*, *toor* and *daemon* share the same home directory. If this directory contains calendar information, **calendar** will process the file three times.

This option requires super-user privileges.

-B *num*

Print lines from today and the previous *num* days (backward, past).

-D *moon*/*sun*

Print UTC offset, longitude and moon or sun information.

-d   Debug option: print current date information.

-F *friday*

Specify which day of the week is ''Friday'' (the day before the weekend begins). Default is 5.

-f *calendarfile*

Use *calendarfile* as the default calendar file.

-l *longitude*

Perform lunar and solar calculations from this longitude. If neither longitude nor UTC offset is specified, the calculations will be based on the difference between UTC time and localtime. If both are specified, UTC offset overrides longitude.

-t *dd*[.*mm*[.*year*]]

For test purposes only: set date directly to argument values.

-U *UTC-offset*

Perform lunar and solar calculations from this UTC offset. If neither UTC offset nor longitude is specified, the calculations will be based on the difference between UTC time and localtime. If both are specified, UTC offset overrides longitude.

-W *num*

Print lines from today and the next *num* days (forward, future). Ignore weekends when calculating the number of days.

FILE FORMAT

To handle calendars in your national code table you can specify "LANG=<locale_name>" in the calendar file as early as possible.

To handle the local name of sequences, you can specify them as: "SEQUENCE=<first> <second> <third> <fourth> <fifth> <last>" in the calendar file as early as possible.

The names of the following special days are recognized:

Easter   Catholic Easter.
Paskha   Orthodox Easter.
NewMoon   The lunar New Moon.
FullMoon   The lunar Full Moon.
MarEquinox   The solar equinox in March.
JunSolstice   The solar solstice in June.
SepEquinox   The solar equinox in September.
DecSolstice   The solar solstice in December.
ChineseNewYear The first day of the Chinese year.

These names may be reassigned to their local names via an assignment like "Easter=Pasen" in the calendar file.

Other lines should begin with a month and day. They may be entered in almost any format, either numeric or as character strings. If the proper locale is set, national month and weekday names can be used. A single asterisk (''*'') matches every month. A day without a month matches that day of every week. A month without a day matches the first of that month. Two numbers default to the month

followed by the day. Lines with leading tabs default to the last entered date, allowing multiple line specifications for a single date.

The names of the recognized special days may be followed by a positive or negative integer, like: "Easter+3" or "Paskha-4".

Weekdays may be followed by ''-4'' ... ''+5'' (aliases for last, first, second, third, fourth) for moving events like ''the last Monday in April''.

By convention, dates followed by an asterisk are not fixed, i.e., change from year to year.

Day descriptions start after the first <tab> character in the line; if the line does not contain a <tab> character, it is not displayed. If the first character in the line is a <tab> character, it is treated as a continuation of the previous line.

The **calendar** file is preprocessed by a limited subset of cpp(1) internally, allowing the inclusion of shared files such as lists of company holidays or meetings. This limited subset consists of **#include**, **#define**, **#undef**, **#ifdef**, **#ifndef**, **#else**, **#warning**, and **#error**.

Conditions can be nested and the consistency of opening and closing instructions is checked. Only the first word after #define is used as the name of the condition variable being defined. More than word following #ifdef, #ifndef, or #undef is considered a syntax error, since names cannot include white-space. Included files are parsed in a global scope with regard to the condition variables being defined or tested therein. All conditional blocks are implicitly closed at the end of a file, and missing #endif instructions are assumed to be present on implied succeeding lines.

If the shared file is not referenced by a full pathname, **calendar** searches in the current (or home) directory first, and then in the directory */usr/share/calendar*.

Blank lines and text protected by the C comment syntax '/* ... */' or '//' are ignored, but the latter only at the beginning of a line or after white space to allow for URLs in calendar entries.

Some possible calendar entries (<tab> characters highlighted by **\t** sequence):

```
LANG=C
Easter=Ostern

#include <calendar.usholiday>
#include <calendar.birthday>
```

6/15\tJune 15 (if ambiguous, will default to month/day).
Jun. 15\tJune 15.
15 June\tJune 15.
Thursday\tEvery Thursday.
June\tEvery June 1st.
15 *\t15th of every month.
2010/4/15\t15 April 2010

May Sun+2\tsecond Sunday in May (Muttertag)
04/SunLast\tlast Sunday in April,
\tsummer time in Europe
Easter\tEaster
Ostern-2\tGood Friday (2 days before Easter)
Paskha\tOrthodox Easter

FILES

calendar   file in current directory.
~/.calendar   *calendar* HOME directory. A chdir is done into this directory if it exists.
~/.calendar/calendar
                        calendar file to use if no calendar file exists in the current directory.
~/.calendar/nomail
                        do not send mail if this file exists.
/usr/share/calendar
                        system wide location of calendar files provided as part of the base system.
/usr/local/share/calendar
                        system wide location for calendar files provided by a port or package.

The order of precedence in searches for a calendar file is: current directory, ~/.calendar,
/usr/local/share/calendar, /usr/share/calendar. Files of similar names are ignored in lower precedence
locations.

The following default calendar files are provided by the deskutils/calendar-data port.

calendar.all   File which includes all the default files.
calendar.australia   Calendar of events in Australia.
calendar.birthday   Births and deaths of famous (and not-so-famous) people.
calendar.christian   Christian holidays. This calendar should be updated yearly by the local system
                        administrator so that roving holidays are set correctly for the current year.
calendar.computer   Days of special significance to computer people.

calendar.croatian   Calendar of events in Croatia.
calendar.dutch   Calendar of events in the Netherlands.
calendar.freebsd   Birthdays of FreeBSD committers.
calendar.french   Calendar of events in France.
calendar.german   Calendar of events in Germany.
calendar.history   Everything else, mostly U.S. historical events.
calendar.holiday   Other holidays, including the not-well-known, obscure, and *really* obscure.
calendar.judaic   Jewish holidays. The entries for this calendar have been obtained from the
                    deskutils/hebcal port.
calendar.music   Musical events, births, and deaths. Strongly oriented toward rock 'n' roll.
calendar.newzealand
                    Calendar of events in New Zealand.
calendar.russian   Russian calendar.
calendar.southafrica Calendar of events in South Africa.
calendar.usholiday   U.S. holidays. This calendar should be updated yearly by the local system
                    administrator so that roving holidays are set correctly for the current year.
calendar.world   Includes all calendar files except for national files.

COMPATIBILITY
The **calendar** program previously selected lines which had the correct date anywhere in the line. This is
no longer true, the date is only recognized when it occurs at the beginning of a line.

SEE ALSO
at(1), mail(1), cron(8)

HISTORY
A **calendar** command appeared in Version 7 AT&T UNIX.

NOTES
Chinese New Year is calculated at 120 degrees east of Greenwich, which roughly corresponds with the
east coast of China. For people west of China, this might result that the start of Chinese New Year and
the day of the related new moon might differ.

The phases of the moon and the longitude of the sun are calculated against the local position which
corresponds with 30 degrees times the time-difference towards Greenwich.

The new and full moons are happening on the day indicated: They might happen in the time period in
the early night or in the late evening. It does not indicate that they are starting in the night on that date.

Because of minor differences between the output of the formulas used and other sources on the Internet,

Druids and Werewolves should double-check the start and end time of solar and lunar events.

BUGS

The **calendar** does only recognise the cpp directives #include, #define, #ifdef, #ifndef and #else. It supports nested conditions, but does not perform any validation on the correct use and nesting of conditions. #endif without prior #ifdef or #define is ignored and #else outside a conditional section skips input lines up to the next #endif.

There is no possibility to properly specify the local position needed for solar and lunar calculations.