

# easter package

## Version 1.0

Florian Tasso\*

July 7, 2026

### Abstract

This package provides commands to compute the date of Easter, Ascension and Pentecost, according to Gregorian and Julian *computus*, implementing the Meeus algorithms. It also provides commands to translate dates between Gregorian and Julian calendars.

## 1 Commands definition

The main commands of this package are `\EasterGregorian`, `\EasterJulian`, `\Ascension` and `\Pentecost`. They are pretty self-explanatory: the first two compute Easter's date with respect to the specified *computus*, and the last two compute the dates of Ascension and Pentecost, which have the same definition in both cases.

They all have the same interface:

- by default, the feast's date is computed for compilation-time year (as stored in `\year`). These commands can also receive an optional argument to select another year, provided it is valid (see section 3),
- these commands store their resulting date in three counters each (`XXXYear`, `XXXMonth` and `XXXDay`, where `XXX` stands for `Easter`, `Ascension` or `Pentecost`), and print it in `YYYY-MM-DD` format. They all have a starred variant to cancel printing: dates can thus be computed at a moment and reused anytime later in the document,
- the last computed date for these feasts may be recalled and printed by the command `\printDate{feast}`, taking the feast's name as a mandatory argument. For instance, using `\EasterGregorian*` followed by `\printDate{Easter}` is equivalent to call `\EasterGregorian`. It is actually how commands do print the date.

Date printing format may be customized manually using the aforementioned counters or thanks to other packages, such as `datetime2`.

A shortcut command, `\Easter`, is also provided, but requires a *computus* to have been previously selected (see section 4).

Seeking efficiency, `\Ascension` and `\Pentecost` commands first check whether Easter date has already been computed. If so, and if the last Easter year matches the requested year for Ascension or Pentecost, `Easter` is not computed again. Otherwise, `\Easter` is invoked first, and consequently needs already to exist.

Further commands are finally defined to convert dates between calendars: see section 3.

## 2 Examples

Dates in the following table are computed by the previous commands. They can be compared with <https://olravet.fr/AideCalendes/ortho.htm#ancre38020> or [https://fr.wikipedia.org/wiki/Calcul\\_de\\_la\\_date\\_de\\_P%C3%A2ques](https://fr.wikipedia.org/wiki/Calcul_de_la_date_de_P%C3%A2ques):

---

\*e-mail: florian dot tasso at gmail dot com

Year	Gregorian Easter		Julian Easter	
	<i>Gregorian Calendar</i>	<i>Julian Calendar</i>	<i>Julian Calendar</i>	<i>Gregorian Calendar</i>
2026	2026-04-05	2026-03-23	2026-03-30	2026-04-12
2025	2025-04-20	2025-04-07	2025-04-07	2025-04-20
2012	2012-04-08	2012-03-26	2012-04-02	2012-04-15
2011	2011-04-24	2011-04-11	2011-04-11	2011-04-24
2034	2034-04-09	2034-03-27	2034-03-27	2034-04-09
2035	2035-03-25	2035-03-12	2035-04-16	2035-04-29

The next command will silently compute the Gregorian date of Easter in 2003: `\EasterGregorian*[2003]`. The result is now displayed by `\printDate{Easter}`: 2003-04-20. This year, Ascension was on `\Ascension[2003]`, *i.e.* 2003-05-29.

Let us now compute the Julian date of Easter for the current year (2026): `\EasterJulian*`. The result can then be displayed in DD/MM/YYYY format with the sequence `\theEasterDay/\theEasterMonth/\theEasterYear`: 30/3/2026<sup>1</sup>. The corresponding date for Pentecost is `\Pentecost*\printDate{Pentecost}`, *i.e.* 2026-05-18.

### 3 Technical Details Regarding Calendars

When dealing with Easter date, one should make distinction between *calendar* and *computus*. The former is about how to name days and group them by months and years: it is the basement of the concept of *date*, which does not make any sense without it. It also defines leap years rules. The later defines how to compute moveable feasts (hence its name), which are based on Easter, depending in turn on the beginning of Spring and Moon phases. These feasts are an important part of Christian religion.

This package allows to compute moveable feasts for both Julian and Gregorian *computus*, and consequently relies on both *calendars*. We refer here to *old* Julian calendar and *computus*: although many Orthodox Churches (but not all) did adopt the new Julian calendar, the new Julian *computus* seems to be almost not used, in favor of the old one.

Computations are made following the Meeus algorithms<sup>2</sup>: Butcher-Meeus for Gregorian *computus*, and Delambre-Meeus for Julian *computus*. These algorithms are implemented using T<sub>E</sub>X counter's arithmetic, thus optimizing compilation time and reducing package dependencies.

Dates for a given *computus* are naturally expressed in the corresponding calendar, but they can be further translated to the other one. It is especially useful to express Orthodox feasts (defined by the Julian *computus*) in Gregorian calendar (more broadly used than Julian one). The opposite operation (expressing Catholic feasts, defined by Gregorian *computus*, in Julian calendar) is also possible, although of a more specific interest.

The difference between Julian and Gregorian calendar is due to discrepancies in Tropical year's approximation, which is about 365.24219 days long. Every four years, Julian calendar add a so called "leap year", lasting 366 days by inserting a 29<sup>th</sup> day in February, to resynchronize calendar and solar year. The mean duration of Julian year is thus 365.25 days, which is better than 365 days, but still a bit long. Julian calendar is therefore slowly sliding after solar year. During the 16<sup>th</sup> Century, the gap was of ten days. The Gregorian calendar was then introduced to compensate this gap and reduce sliding rate, by removing three leap years every four centuries. This calendar's year is then 365.2425 days long. If it is still not perfect, it is at least more accurate than old Julian calendar. The revised Julian calendar is even more accurate with a mean year of about 365.24222 days, but it will not be discussed here.

The gap between (old) Julian and Gregorian calendars for a given year can be easily computed. The package offers the command `\calendarOffset` for this purpose: it takes the desired year as an optional argument (or by default the compilation-time year), and set the counter `calendaroffset` to the right value. For instance, when this document was compiled (in 2026), Gregorian calendar was 13 days before Julian one.

Based on this offset, one can define commands to "translate" a date from one calendar to the other. It is the purpose of `\toGregorian{year}{month}{day}` and `\toJulian{year}{month}{day}`, which set the counters

<sup>1</sup>Actually, displaying a counter's value using `\theCOUNTERNAME` does not print leading 0. To do so, one may use the L<sup>A</sup>T<sub>E</sub>X macro `\two@digits` or implement an *ad hoc* function.

<sup>2</sup>See [https://fr.wikipedia.org/wiki/Calcul\\_de\\_la\\_date\\_de\\_P%C3%A2ques\\_selon\\_la\\_m%C3%A9thode\\_de\\_Meeus](https://fr.wikipedia.org/wiki/Calcul_de_la_date_de_P%C3%A2ques_selon_la_m%C3%A9thode_de_Meeus).

`TransposedYear`, `TransposedMonth` and `TransposedDay` to the correct values. The resulting date may then be printed using `\printDate{Transposed}`, or by manipulating those counters directly.

## 4 Package Options

This package supports two options: `gregorian` and `julian`. Their effect is to define `\Easter` as an alias for the specified *computus*.

*Computus* can also be selected manually with `\selectComputus` command. It takes one mandatory argument, `julian` or `gregorian`, and set `\Easter` accordingly. Actually, it is how options are handled.

Note that this alias is *not* defined if the package is loaded without option, or more generally before the previous command is invoked. This behaviour is intended to remain *computus*-agnostic, event though Gregorian calendar is the most widespread.

## 5 Package History

**v1.0** – 2026/07/07: Initial release.