The `isodate` package*

Harald Harders
h.harders@tu-bs.de

File Date 2001-05-04, Printed 2001-05-17

Abstract

This package provides commands to switch between different date formats (standard, ISO, numeric, LaTeX package). They are used by the `\today` command, by the `\printdate` and `\printdateTeX` commands that print any date, and by the `\daterange` command that prints a date range. At the moment this package supports German (old and new rules, Austrian), British English, US English, French, Danish, and Norwegian.

The idea for this package was taken from the akletter class.

Contents

*This file has version 2.04 last revised 2001-05-04, documentation dated 2001-05-17.

Acknowledgements

First of all I have to thank Axel Kielhorn who wrote the package `akletter` which inspired me to write `isodate`. The help of Heiko Oberdiek was necessary to handle characters in substrings which resulted in the package `substr`. David Sanderson found the bug which disabled `isodate` to work without `babel`. He also helped me to improve the documentation and sent me a link to the ISO 8601 norm [1]. Svend Tollak Munkejord has added the norwegian language.

1 Commands

1.1 Switching the date format

This package provides five commands to switch the output format of all commands that print dates (described later):

\isodate
\numdate
\shortdate
\TeXdate
\origdate

`\isodate`	date format described in ISO 8601 and DIN 5008 [1] (yyyy-mm-dd)
`\numdate`	numeric date format with four digits of the year
`\shortdate`	short numeric date format with two digits of the year
`\TeXdate`	date format used for version description of packages (yyyy/mm/dd)
`\origdate`	original LaTeX format

The numeric and short numeric format change their behaviour depending on the actual language:

German, nGerman	`dd.\,mm.~yyyy`	resp.	`dd.\,mm.\,yy`
US English	`mm/dd/yyyy`	resp.	`mm/dd/yy`
other languages	`dd/mm/yyyy`	resp.	`dd/mm/yy`

This package supports German (old and new rules, Austrian), US English, French, Danish, and Norwegian. Switching the language by using `\selectlanguage` does *not* switch back to the original date format. So the current date format stays active when changing the language.

The change of the date format works locally. So it is possible to change it locally inside a group. E.g.

```
\today, {\origdate\today}, \today
```

leads to “2001-05-17, 17th May 2001, 2001-05-17”.

1.2 Printing today's date

\today As usual the command `\today` prints the date of today. Its appearance is influenced by the current date format

1.3 Printing any date

\printdate The command `\printdate{#1}` prints any date in the current format. The argument may be a date in German, British English, or ISO format, e.g.

```
\printdate{24.12.2000}
\printdate{24/12/2000}
\printdate{2000-12-24}
```

\printdateTeX The command `\printdateTeX{#1}` prints any date in the actual format. The argument must be in the LaTeX format `yyyy/mm/dd`, e.g.

```
\printdateTeX{2000/12/24}
```

This command is useful for printing version information stored in a macro. For example the version of this documentation is stored in the macro `\docdate` ("2001/05/17"). To print it with the actual date format you can use the command `\printdateTeX{\docdate}` which leads to e.g. "2001-05-17" or "17th May 2001".

1.4 Printing date ranges

\daterange The command `\daterange{#1}{#2}` prints a date range in the current format. The arguments may be a date in German, British English, or ISO format (see above). But there is a limitation: Both arguments must have the same input format.

Depending on the language and date format this commands lets out some of the data. The simplest way to understand it is to watch some examples:

`{\isodate`
`\daterange{1999-05-03}{1999-05-31}` $\longrightarrow$ 1999-05-03 to 31
`\daterange{1999-05-03}{1999-11-03}` $\longrightarrow$ 1999-05-03 to 11-03
`\daterange{1999-05-03}{2000-04-07}` $\longrightarrow$ 1999-05-03 to 2000-04-07
`}`
`{\origdate`
`\daterange{1999-05-03}{1999-05-31}` $\longrightarrow$ 3rd to 31st May 1999
`\daterange{1999-05-03}{1999-11-03}` $\longrightarrow$ 3rd May to 3rd November 1999
`\daterange{1999-05-03}{2000-04-07}` $\longrightarrow$ 3rd May 1999 to 7th April 2000
`}`

1.5 Changing the ISO format

\isodash The ISO norm says that the date format is "yyyy-mm-dd" or "yyyymmdd" [1].

By default I use the hyphen "-" as separator. You can change this using the `\isodash`[1] command, e.g.

```
\printdate{24/12/2000},
\isodash{--}%
\printdate{24/12/2000},
\isodash{}%
\printdate{24/12/2000}
```

leads to "2000-12-24, 2000–12–24, 20001224". Or for example

```
\isodash{$\cdot$}
\printdate{24/12/2000}
```

leads to "2000·12·24".

1.6 Changing the German format

`\daymonthsepgerman`
`\monthyearsepgerman`

The spacings for the numerical formats in the German language (24. 12. 2000 resp. 24. 12. 00) was taken from the Duden [2] and is the default spacing when using one of the German derivatives. Some people want to use different spacings. Thus from version 2.03 on it will be possible to change it. You can change the spacing between the day and the month using the command `\daymonthsepgerman`. Using the command `\monthyearsepgerman` you can change the spacing between the month and the year for the long and the short format, e.g.

```
\daymonthsepgerman{\quad}%
\monthyearsepgerman{\qquad}{\quad}%
{\numdate\printdate{24.12.2000}}, {\shortdate\printdate{24.12.2000}}
```

leads to "24. 12.   2000, 24. 12. 00".

2 Calling the package

The package is called using the `\usepackage` command:
`\usepackage[option]{isodate}`.

The possible package options can be seen in table 1.

Be aware that at least one language option must be set when calling isodate. The last language in the option list is the default language.

The package isodate works well together with `babel.sty`, `german.sty`, or `ngerman.sty`. Maybe it also works together with other language packages like e.g. `french.sty`, but this has not been tested. It does not matter if `isodate` is loaded before or after the used language package.

[1] The name "isodash" is a little bit confusing and was chosen due to my limited knowledge in English. It should be named "isoseparator" or "isosep". But for compatiblity reasons I will not change it.

Table 1: Package options

option	function
iso	start with ISO date format
num	start with numeric date format with 4 digits of the year
short	start with numeric date format with 2 digits of the year
TeX	start with LaTeX numeric date format (yyyy/mm/dd)
orig	start with normal LaTeX date format (default)
american	support American English date format
austrian	support Austrian date format
danish	support Danish date format
english	support British English date format
french	support French date format
german	support German date format
naustrian	support new Austrian date format
ngerman	support new German date format
norsk	support Norwegian date format
norwegian	support Norwegian date format

It is also possible to use isodate without one of the language packages. Then it is not possible to switch between languages using the `\selectlanguage` command.[2] Then the default language is the last one in the option list. If an error occurs when using isodate without one of the packages `babel.sty`, `german.sty`, and `ngerman.sty` please run `tstlang.tex` through latex and send the file `tstlang.log` to the address `h.harders@tu-bs.de`.

If using isodate together with babel it can be useful to put the language options as global options into the optional parameters of the `\documentclass` command. Then automatically the available languages are the same for the text and the dates, and the default language is also the same. For example:

```
\documentclass[english,german]{article}
\usepackage{babel}
\usepackage[num]{isodate}
```

3 Add new languages to the package

The easiest way to add new languages to the package is to copy one of the simple language files `danish.idf` or `french.idf` to the new language name, e.g. `plattdeutsch.idf`, and change it as necessary.

This new file can be used without changing `isodate.sty` if you use its name explicitly in the optional parameter of the `\usepackage` command. If you have added support for a new language please mail me.

[2] Yes, there is a way to change the date language, but it is a little bit tricky:

```
\makeatletter
\def\iso@languagename{german}%
\dategerman%
\makeatother
```

4 Solvable problems

4.1 Redefinition of language dependent names

If someone wants to change one of the language dependent names (e.g. `\contentsname`, `\figurename`, ...) many people just write (e.g. in the language `ngerman`)

```
\renewcommand{\contentsname}{Inhalt}
```

in the preamble of the document. If they are using `german.sty`, `ngerman.sty`, or no language package (and not `isodate.sty`) this approach works. But when using `babel.sty` and/or `isodate.sty` the default ("Inhaltsverzeichnis") results. This is due to the fact that these packages call the command `\selectlanguage` at the `\begin{document}`.

Depending on the used package different solutions have to be used.

Using `babel.sty` the redefinition of the name has to be added to the language extras:

```
\addto{\extrasngerman}{\renewcommand{\contentsname}{Inhalt}}
```

Using `german.sty` or `ngerman.sty` this approach does not work. Instead, the `\selectlanguage` command has to be redefined. First, the original command has to be copied via the `\let` command. Then the `\selectcommand` is redefined using the original command:

```
\let\origselectlanguage=\selectlanguage
\def\selectlanguage#1{\origselectlanguage{#1}%
  \iflanguage{ngerman}{\renewcommand{\contentsname}{Inhalt}}{%
  \iflanguage{german}{\renewcommand{\contentsname}{Inhalt}}{}}}
 \selectlanguage{ngerman}
```

These commands should be placed shortly before the `\begin{document}` command.

A Licence

B Known errors

- The `\printdate` and `\printdateTeX` commands are not very good in checking the argument for correct syntax.
- I don't know the French word that should be used in a date range. So I have taken `~--~` (" – ").
- The language definition files `french.idf` and `german.idf` are not yet commented.

C Planned features and changes

- Of course eliminate the errors.
- Add other languages than German, new German, English, US English, French, Danish, and Norwegian. Please help me with this topic. I don't know the date formats in other languages.

D Literature

References

[1] International Standard: ISO 8601. http://www.iso.ch/markete/8601.pdf, 1988-06-15.

[2] DUDEN Band 1. Die deutsche Rechtschreibung. 21. Auflage, Dudenverlag, Mannheim, Germany, 1996.

E The implementation

E.1 Package file isodate.sty

Heading of the package:

```
\NeedsTeXFormat{LaTeX2e}
\ProvidesPackage{isodate}[\filedate\space v\fileversion
  Change the output format of dates (HH)]
\RequirePackage{ifthen}
\RequirePackage{calc}
\IfFileExists{substr.sty}{\RequirePackage{substr}%
 }{\PackageError{isodate.sty}{Package file substr.sty not found}
   {This version of isodate.sty needs the package substr.sty.^^J%
     You can download it from
     CTAN:/macros/latex/contrib/supported/substr/^^J%
     E.g. one CTAN node is ftp.dante.de.
     Install substr.sty into your TeX tree.}}
```

Declare the options for the default date format.

```
\DeclareOption{iso}{\AtBeginDocument{\isodate}}
\DeclareOption{num}{\AtBeginDocument{\numdate}}
\DeclareOption{short}{\AtBeginDocument{\shortdate}}
\DeclareOption{TeX}{\AtBeginDocument{\TeXdate}}
\DeclareOption{orig}{\AtBeginDocument{\origdate}}
```

Declare the options for language support.

```
\DeclareOption{austrian}{\input{german.idf}}
\DeclareOption{english}{\input{english.idf}}
\DeclareOption{german}{\input{german.idf}}
\DeclareOption{american}{\input{english.idf}}
\DeclareOption{naustrian}{\input{german.idf}}
\DeclareOption{ngerman}{\input{german.idf}}
\DeclareOption{UKenglish}{\input{english.idf}}
\DeclareOption{USenglish}{\input{english.idf}}
```

```
\DeclareOption{danish}{\input{danish.idf}}
\DeclareOption{norsk}{\input{norsk.idf}}
\DeclareOption{norwegian}{\input{norsk.idf}}
\DeclareOption{french}{\input{french.idf}}
```

Make it possible to load language definition files that are not known by this package.

```
\DeclareOption*{%
  \InputIfFileExists{\CurrentOption.idf}{}{%
    \PackageError{isodate}{%
      Isodate definition file \CurrentOption.idf not found}{%
      Maybe you misspelled the language option?}}%
  }
```

Set default option to `orig`.

```
\ExecuteOptions{orig}
```

Process the options.

```
\ProcessOptions*
```

Handle the case that no language was given. Throw an error message. Each language definition file `*.idf` must contain a line

```
\let\iso@languageloaded\active
```

that defines the command `\iso@languageloaded`.

```
\ifx\iso@languageloaded\@undefined
  \PackageError{isodate}{%
    You haven't specified a language option}{%
    You need to specify a language, either as a global
    option\MessageBreak
    or as an optional argument to the \string\usepackage\space
    command.\MessageBreak
    If you have used the old isodate package (version <=1.06) you can
    change the\MessageBreak
    usepackage command to \protect\usepackage{isodate}.\MessageBreak
    You shouldn't try to proceed from here, type x to quit.}
\fi
```

`\printnumber` Prints a number. If the optional parameter is used the number has two digits and it is filled by “0”.

Examples: `\printnumber{4}` ⟶ 4, `\printnumber[g]{4}` ⟶ 04.

```
\newcommand{\printnumber}[2][\@empty]{%
  \ifthenelse{\equal{#1}{\@empty}}{}{%
    \ifthenelse{\number#2<10}{0}{}}\number#2}%
```

`\iso@yeartwo` Prints the argument of the command with two digits.

Example: `\iso@yeartwo{1873}` ⟶ 73.

```
\newcounter{iso@yeartwo}%
\newcommand{\iso@yeartwo}[1]{%
  \setcounter{iso@yeartwo}{\number#1}%
  \whiledo{\theiso@yeartwo>99}{%
    \setcounter{iso@yeartwo}{\theiso@yeartwo-100}}{}%
  \ifthenelse{\number\theiso@yeartwo<10}{0}{}\theiso@yeartwo}%
```

\iso@dateformat In this command the current active date format ist stored. Possible values are: `numeric`, `short`, `iso`, `orig`, `TeX`.

```
\def\iso@dateformat{numeric}
```

\numdate Switches to long numerical date format.

```
\DeclareRobustCommand*{\numdate}{\def\iso@dateformat{numeric}}%
```

\shortdate Switches to short numerical date format.

```
\DeclareRobustCommand*{\shortdate}{\def\iso@dateformat{short}}%
```

\isodate Switches to ISO date format.

```
\DeclareRobustCommand*{\isodate}{\def\iso@dateformat{iso}}%
```

\origdate Switches to the original date format.

```
\DeclareRobustCommand*{\origdate}{\def\iso@dateformat{orig}}%
```

\TeXdate Switches to LaTeX date format.

```
\DeclareRobustCommand*{\TeXdate}{\def\iso@dateformat{TeX}}%
```

\isodash Changes the dash in the ISO date format. The default is “-”.

```
\def\iso@isodash{-}%
\DeclareRobustCommand*{\isodash}[1]{\def\iso@isodash{#1}}%
```

\isorangesign Defines the sign or word that is printed between the two dates in a date range. E.g. in English the default is “ to ”.

```
\def\iso@rangesign{\csname iso@rangesign@\iso@languagename\endcsname}%
\DeclareRobustCommand*{\isorangesign}[1]{\def\iso@rangesign{#1}}%
```

\iso@printdate Defines the command `iso@printdate` which takes three arguments (year, month, day) and prints the date by using the `\today` command.

```
\newcommand{\iso@printdate}[3]{%
  \begingroup%
  \def\year{#1}%
  \def\month{#2}%
  \def\day{#3}%
  \today%
  \endgroup%
}
```

\printdate Prints a date that is given as one argument in one of these formats: `yyyy-mm-dd`, `dd/mm/yyyy`, `dd.mm.yyyy`.

```
\DeclareRobustCommand*{\printdate}[1]{%
```

Define `\iso@date` command to expand the argument `#1`.

```
  \edef\iso@date{#1}%
```

Count appearances of “/”, “-”, and “.” in the argument.

```
  \SubStringsToCounter{iso@slash}{/}{\iso@date}%
  \SubStringsToCounter{iso@minus}{-}{\iso@date}%
  \SubStringsToCounter{iso@dot}{.}{\iso@date}%
```

If number of “.” in the argument is equal to 2 then the German format `dd.mm.yyyy` is used.

```
  \ifthenelse{\equal{\theiso@dot}{2}}{%
    \expandafter\iso@input@german\iso@date\@empty}{%
```

If number of "-" in the argument is equal to 2 then the ISO format `yyyy-mm-dd` is used.

```
  \ifthenelse{\equal{\theiso@minus}{2}}{%
    \expandafter\iso@input@iso\iso@date\@empty}{%
```

If number of "/" in the argument is equal to 2 then the British English format `dd/mm/yyyy` is used.

```
    \ifthenelse{\equal{\theiso@slash}{2}}{%
      \expandafter\iso@input@english\iso@date\@empty}{%
```

Else no of the formats above is used an thus an error message is thrown.

```
      ????\iso@isodash ??\iso@isodash ??%
      \PackageError{isodate}{unrecognized date format}{Use one of
        the following formats as macro argument:^^J%
        \space\space dd.mm.yyyy^^J%
        \space\space dd/mm/yyyy^^J%
        \space\space yyyy-mm-dd^^J%
        Don't use any spaces or commands like \protect\, or
        \protect~ inside the argument.}%
      }}}%
}
```

`\iso@input@iso` Converts a string with the format `yyyy-mm-dd` to three arguments `{#1}{#2}{#3}` and calls `\iso@printdate`.

```
\def\iso@input@iso#1-#2-#3\@empty{\iso@printdate{#1}{#2}{#3}}
```

`\iso@input@german` Converts a string with the format `dd.mm.yyyy` to three arguments `{#3}{#2}{#1}` and calls `\iso@printdate`.

```
\def\iso@input@german#1.#2.#3\@empty{\iso@printdate{#3}{#2}{#1}}
```

`\iso@input@english` Converts a string with the format `dd/mm/yyyy` to three arguments `{#3}{#2}{#1}` and calls `\iso@printdate`.

```
\def\iso@input@english#1/#2/#3\@empty{\iso@printdate{#3}{#2}{#1}}
```

`\printdateTeX` Prints a date that is given as one argument in the format `yyyy/mm/dd`.

```
\DeclareRobustCommand*{\printdateTeX}[1]{%
```

Define `\iso@date` command to expand the argument `#1`.

```
  \edef\iso@date{#1}%
```

Count appearances of "/" in the argument.

```
  \SubStringsToCounter{iso@slash}{/}{\iso@date}%
```

If number of "/" in the argument is equal to 2 then the LaTeX format `yyyy/mm/dd` is used.

```
  \ifthenelse{\equal{\theiso@slash}{2}}{%
    \expandafter\iso@input@TeX\iso@date\@empty}{%
```

Else no of the formats above is used an thus an error message is thrown.

```
    ????\iso@isodash ??\iso@isodash ??%
    \PackageError{isodate}{unrecognized date format}{Use one of
      the following formats as macro argument:^^J%
      \space\space dd.mm.yyyy^^J%
      \space\space dd/mm/yyyy^^J%
      \space\space yyyy-mm-dd^^J%
```

```
      Don't use any spaces or commands like \protect\, or
      \protect~ inside the argument.}%
    }}
```

`\iso@input@TeX` Converts a string with the format `yyyy/mm/dd` to three arguments `{#1}{#2}{#3}` and calls `\iso@printdate`.

```
\def\iso@input@TeX#1/#2/#3\@empty{\iso@printdate{#1}{#2}{#3}}
```

`\daterange` Prints a date range.

```
\DeclareRobustCommand*{\daterange}[2]{%
```

Define `\iso@date` and `\iso@@date` commands to expand the argument `#1` and `#2`. Define `\iso@@@date` which contains both arguments devided by a komma.

```
  \edef\iso@date{#1}%
  \edef\iso@@date{#2}%
  \edef\iso@@@date{\iso@date,\iso@@date}%
```

Count appearances of "/", "-", and "." in the arguments.

```
  \SubStringsToCounter{iso@slash}{/}{\iso@date}%
  \SubStringsToCounter{iso@minus}{-}{\iso@date}%
  \SubStringsToCounter{iso@dot}{.}{\iso@date}%
  \SubStringsToCounter{iso@@slash}{/}{\iso@@date}%
  \SubStringsToCounter{iso@@minus}{-}{\iso@@date}%
  \SubStringsToCounter{iso@@dot}{.}{\iso@@date}%
```

If number of "." in both arguments is equal to 2 then the German format `dd.mm.yyyy` is used.

```
  \ifthenelse{\equal{\theiso@dot}{2}\and\equal{\theiso@@dot}{2}}{%
    \expandafter\iso@range@input@german\iso@@@date\@empty}{%
```

If number of "-" in both arguments is equal to 2 then the ISO format `yyyy-mm-dd` is used.

```
    \ifthenelse{\equal{\theiso@minus}{2}\and\equal{\theiso@@minus}{2}}{%
      \expandafter\iso@range@input@iso\iso@@@date\@empty}{%
```

If number of "/" in both arguments is equal to 2 then the British English format `dd/mm/yyyy` is used.

```
      \ifthenelse{\equal{\theiso@slash}{2}\and%
                  \equal{\theiso@@slash}{2}}{%
        \expandafter\iso@range@input@english\iso@@@date\@empty}{%
```

Else no of the formats above is used an thus an error message is thrown.

```
        ????\iso@isodash ??\iso@isodash ??%
        \PackageError{isodate}{unrecognized date format}{Use one of
          the following formats as macro argument:^^J%
          \space\space dd.mm.yyyy^^J%
          \space\space dd/mm/yyyy^^J%
          \space\space yyyy-mm-dd^^J%
          Don't use any spaces or commands like \protect\, or
          \protect~ inside the argument.^^J
          Use the same format for both arguments.}%
        }}}%
}
```

\iso@range@input@iso Converts a string with the format yyyy-mm-dd,yyyy-mm-dd to six arguments {#1}{#2}{#3}{#4}{#5}{#6} and calls \iso@daterange@language.

```
\def\iso@range@input@iso#1-#2-#3,#4-#5-#6\@empty{%
  \csname iso@daterange@\iso@languagename\endcsname{%
    #1}{#2}{#3}{#4}{#5}{#6}}
```

\iso@range@input@german Converts a string with the format dd.mm.yyyy,dd.mm.yyyy to six arguments {#3}{#2}{#1}{#6}{#5}{#4} and calls \iso@daterange@language.

```
\def\iso@range@input@german#1.#2.#3,#4.#5.#6\@empty{%
  \csname iso@daterange@\iso@languagename\endcsname{%
    #3}{#2}{#1}{#6}{#5}{#4}}
```

\iso@range@input@english Converts a string with the format dd/mm/yyyy,dd/mm/yyyy to six arguments {#3}{#2}{#1}{#6}{#5}{#4} and calls \iso@daterange@language.

```
\def\iso@range@input@english#1/#2/#3,#4/#5/#6\@empty{%
  \csname iso@daterange@\iso@languagename\endcsname{%
    #3}{#2}{#1}{#6}{#5}{#4}}
```

Define the counters for conting the appearances of ".", "-", and "/" in the arguments.

```
\newcounter{iso@slash}
\newcounter{iso@minus}
\newcounter{iso@dot}
\newcounter{iso@@slash}
\newcounter{iso@@minus}
\newcounter{iso@@dot}
```

The command \iso@languagename is defined to be able to use this package without loading one of the language packages babel.sty, german.sty, or ngerman.sty.

If neither babel.sty nor german.sty nor ngerman.sty is loaded my computer returns "nohyphenation" when using \languagename. So this is the indication that none of the above packages is loaded.

```
\AtBeginDocument{%
  \ifthenelse{\not\equal{\languagename}{nohyphenation}}{%
```

The language is not equal "nohyphenation". So one of the language packages is loaded. Replace the internal language name \iso@languagename by the global language name \languagename.

```
    \gdef\iso@languagename{\languagename}%
```

Reload language to surely switch to new date format. The languagename gets first expanded because of errors that would occur otherwise.

```
    \edef\iso@tmplang{\languagename}%
    \expandafter\selectlanguage\expandafter{\iso@tmplang}%
  }{%
```

None of the language packages are loaded. So no language switching is possible. Set the date language manually to the last language that was loaded for isodate.

```
    \csname date\iso@languagename\endcsname%
  }
}
```

The end of the package.

E.2 Language definition file danish.idf

Heading of the file:

```
\ProvidesFile{danish.idf}[\filedate\space v\fileversion\space
  Danish support from the isodate package]
```

`\iso@languageloaded` Define the command `\iso@languageloaded` in order to enable `isodate.sty` to determine if at least one language is loaded.

```
\let\iso@languageloaded\active
```

`\month@danish` Prints the name of today's month in the long form for the original date format.

```
\def\month@danish{\ifcase\month\or
    januar\or februar\or marts\or april\or maj\or juni\or
    juli\or august\or september\or oktober\or november\or december\fi}
```

`\iso@printmonthday@danish` Prints the month and the day given as two arguments (`{mm}{dd}`) in the current date format.

```
  \def\iso@printmonthday@danish#1#2{%
```

Numeric and short date format: `dd/mm/`

```
    \ifthenelse{\equal{\iso@dateformat}{numeric}\or%
      \equal{\iso@dateformat}{short}}{%
      \printnumber[10]{#2}/\printnumber[10]{#1}/}{%
```

ISO date format: `-mm-dd`

```
      \ifthenelse{\equal{\iso@dateformat}{iso}}{%
        \iso@isodash\printnumber[10]{#1}%
        \iso@isodash\printnumber[10]{#2}}{%
```

LaTeX date format: `/mm/dd`

```
        \ifthenelse{\equal{\iso@dateformat}{TeX}}{%
          /\printnumber[10]{#1}/\printnumber[10]{#2}}{%
```

Original date format: `d. mmm`

```
          \ifthenelse{\equal{\iso@dateformat}{orig}}{%
            \printnumber{#2}.~\begingroup
            \edef\lmonth{#1}\def\month{\lmonth}%
            \month@danish%
            \endgroup
            }{}}}}%
    }
```

`\iso@printdate@danish` Prints the date given as three arguments (`{yyyy}{mm}{dd}`) in the actual date format

```
  \def\iso@printdate@danish#1#2#3{%
```

ISO or LaTeXdate format: `yyyy\iso@printmonthday@danish`

```
    \ifthenelse{\equal{\iso@dateformat}{iso}\or%
      \equal{\iso@dateformat}{TeX}}{\number#1}{}%
    \iso@printmonthday@danish{\number#2}{\number#3}%
```

numeric date format: `\iso@printmonthday@danish yyyy`

```
    \ifthenelse{\equal{\iso@dateformat}{numeric}}{\number#1}{%
```

original date format: `\iso@printmonthday@danish~yyyy`

```
      \ifthenelse{\equal{\iso@dateformat}{orig}}{~\number#1}{%
```

short date format: `\iso@printmonthday@danish yy`

```
        \ifthenelse{\equal{\iso@dateformat}{short}}{%
          \iso@yeartwo{\number#1}}{%
          }}}%
    }
```

`\iso@datedanish` This command redefines the `\today` command to print in the actual date format.

```
  \def\iso@datedanish{%
    \def\today{\iso@printdate@danish{\year}{\month}{\day}}}%
```

`\iso@daterange@danish` This command takes six arguments (`{yyyy1}{mm1}{dd1}{yyyy2}{mm2}{dd2}`) and prints the corrosponding date range in the actual date format.

```
\def\iso@daterange@danish#1#2#3#4#5#6{%
```

ISO or LaTeX date format.

```
  \ifthenelse{\equal{\iso@dateformat}{iso}\or%
              \equal{\iso@dateformat}{TeX}}{%
```

Print the start date.

```
    \csname iso@printdate@\iso@languagename\endcsname{%
      #1}{#2}{#3}\iso@rangesign%
```

If year and month are equal, only print the day of the end date. If only the year is equal, only print month and day of the end date. Otherwise print the whole end date.

```
    \ifthenelse{\equal{\number#1}{\number#4}}{%
      \ifthenelse{\equal{\number#2}{\number#5}}{\printnumber[10]{#6}%
        }{\iso@printmonthday@danish{#5}{#6}}}{%
      \csname iso@printdate@\iso@languagename\endcsname{#4}{#5}{#6}}}{%
```

Numeric, short, or original date format.

If year and month are equal, only print the day of the start date. If only the year is equal, only print month and day of the start date. Otherwise print the whole start date.

```
    \ifthenelse{\equal{\number#1}{\number#4}}{%
      \ifthenelse{\equal{\number#2}{\number#5}}{%
        \ifthenelse{\equal{\iso@dateformat}{orig}}{%
          \printnumber{#3}.}{\printnumber[10]{#3}}%
        }{\iso@printmonthday@danish{#2}{#3}}}{%
      \csname iso@printdate@\iso@languagename\endcsname{%
        #1}{#2}{#3}}%
```

Print the end date.

```
    \iso@rangesign\csname iso@printdate@\iso@languagename\endcsname{%
      #4}{#5}{#6}%
}{%
    }%
}
```

`\iso@rangesign@danish` Sets the word between start and end date in a date range to " til ".

```
\def\iso@rangesign@danish{~til~}
```

Define the language name that will the active language for isodate if none of the packages babel.sty, german.sty, and ngerman.sty is loaded and if this is the last language that is used for isodate. If one of the above packages is used this definition will be overridden by the command `\languagename` that will always return the current used language.

```
\def\iso@languagename{danish}%
```

Redefine the command `\datedanish` that is used by babel to switch to the original Danish date format to enable the use of different date formats.

```
\AtBeginDocument{%
  \ifx\undefined\iso@datedanish\else
    \def\datedanish{\iso@datedanish}%
  \fi
}
```

The end of the file.

E.3 Language definition file english.idf

Heading of the file:

```
\ProvidesFile{english.idf}[\filedate\space v\fileversion\space
  English support from the isodate package]
```

`\iso@languageloaded` Define the command `\iso@languageloaded` in order to enable `isodate.sty` to determine if at least one language is loaded.

```
\let\iso@languageloaded\active
```

`\month@english` Prints the name of today's month in the long form for the original date format.

```
\def\month@english{\ifcase\month\or
    January\or February\or March\or April\or May\or June\or
    July\or August\or September\or October\or November\or December\fi}
```

British and American English dates are very different. So handle them seperately. It might have been easier to put them in different files but I wanted to organize my files analogous to babel.

First handle British English.

```
\ifthenelse{\equal{\CurrentOption}{english}\or%
            \equal{\CurrentOption}{UKenglish}}{%
```

`\day@english` Prints today's day for the original date format.

```
  \def\day@english{\ifcase\day\or
    1st\or 2nd\or 3rd\or 4th\or 5th\or
    6th\or 7th\or 8th\or 9th\or 10th\or
    11th\or 12th\or 13th\or 14th\or 15th\or
    16th\or 17th\or 18th\or 19th\or 20th\or
    21st\or 22nd\or 23rd\or 24th\or 25th\or
    26th\or 27th\or 28th\or 29th\or 30th\or
    31st\fi}
```

`\iso@printmonthday@english` Prints the month and the day given as two arguments (`{mm}{dd}`) in the current date format.

```
  \def\iso@printmonthday@english#1#2{%
```

Numeric and short date format: `dd/mm/`

```
    \ifthenelse{\equal{\iso@dateformat}{numeric}\or%
      \equal{\iso@dateformat}{short}}{%
      \printnumber[10]{#2}/\printnumber[10]{#1}/}{%
```

ISO date format: mm-dd

```
    \ifthenelse{\equal{\iso@dateformat}{iso}}{%
      \printnumber[10]{#1}\iso@isodash\printnumber[10]{#2}}{%
```

LaTeX date format: mm/dd

```
      \ifthenelse{\equal{\iso@dateformat}{TeX}}{%
        \printnumber[10]{#1}/\printnumber[10]{#2}}{%
```

Original date format: ddd mmm

```
        \ifthenelse{\equal{\iso@dateformat}{orig}}{%
          \begingroup
          \edef\lday{#2}\def\day{\lday}%
          \edef\lmonth{#1}\def\month{\lmonth}%
          \day@english~\month@english%
          \endgroup
          }{}}}}%
  }
```

`\iso@printdate@english` Prints the date given as three arguments ({yyyy}{mm}{dd}) in the actual date format.

```
\def\iso@printdate@english#1#2#3{%
```

ISO date format: yyyy-\iso@printmonthday@english

```
  \ifthenelse{\equal{\iso@dateformat}{iso}}{\number#1\iso@isodash}{%
```

LaTeX date format: yyyy/\iso@printmonthday@english

```
    \ifthenelse{\equal{\iso@dateformat}{TeX}}{\number#1/}{}}%
  \iso@printmonthday@english{\number#2}{\number#3}%
```

Numeric date format: \iso@printmonthday@english yyyy

```
  \ifthenelse{\equal{\iso@dateformat}{numeric}}{\number#1}{%
```

Original date format: \iso@printmonthday@english~yyyy

```
    \ifthenelse{\equal{\iso@dateformat}{orig}}{~\number#1}{%
```

Short date format: \iso@printmonthday@english yy

```
      \ifthenelse{\equal{\iso@dateformat}{short}}{%
        \iso@yeartwo{\number#1}}{%
        }}}%
  }
```

`\iso@printdate@UKenglish` Just a second name for \iso@printdate@UKenglish.

```
\def\iso@printdate@UKenglish{\iso@printdate@english}
```

`\iso@dateenglish` This command redefines the \today command to print in the actual date format.

```
\def\iso@dateenglish{%
  \def\today{\iso@printdate@english{\year}{\month}{\day}}}%
```

`\iso@daterange@english` This command takes six arguments ({yyyy1}{mm1}{dd1}{yyyy2}{mm2}{dd2}) and prints the corrosponding date range in the actual date format.

```
\def\iso@daterange@english#1#2#3#4#5#6{%
```

ISO or LaTeX date format.

```
\ifthenelse{\equal{\iso@dateformat}{iso}\or%
            \equal{\iso@dateformat}{TeX}}{%
```

Print the start date.

```
    \csname iso@printdate@\iso@languagename\endcsname{%
      #1}{#2}{#3}\iso@rangesign%
```

If year and month are equal, only print the day of the end date. If only the year is equal, only print month and day of the end date. Otherwise print the whole end date.

```
    \ifthenelse{\equal{\number#1}{\number#4}}{%
      \ifthenelse{\equal{\number#2}{\number#5}}{\printnumber[10]{#6}%
        }{\iso@printmonthday@english{#5}{#6}}}{%
      \csname iso@printdate@\iso@languagename\endcsname{#4}{#5}{#6}}}{%
```

Numeric, short, or original date format.

If year and month are equal, only print the day of the start date. If only the year is equal, only print month and day of the start date. Otherwise print the whole start date.

```
    \ifthenelse{\equal{\number#1}{\number#4}}{%
      \ifthenelse{\equal{\number#2}{\number#5}}{%
        \ifthenelse{\equal{\iso@dateformat}{orig}}{%
          \begingroup
            \edef\lday{#3}\def\day{\lday}%
            \day@english\endgroup}{\printnumber[10]{#3}}%
        }{\iso@printmonthday@english{#2}{#3}}}{%
      \csname iso@printdate@\iso@languagename\endcsname{#1}{#2}{#3}}%
```

Print the end date.

```
    \iso@rangesign\csname iso@printdate@\iso@languagename\endcsname{%
      #4}{#5}{#6}%
}{%
    }%
}
```

`\iso@daterange@english` Sets the word between start and end date in a date range to " to".

```
\def\iso@rangesign@english{~to~}
```

Define the language name that will the active language for isodate if none of the packages babel.sty, german.sty, and ngerman.sty is loaded and if this is the last language that is used for isodate. If one of the above packages is used this definition will be overridden by the command `\languagename` that will always return the current used language.

```
\def\iso@languagename{english}%
```

The end of the British section.

Second handle American English.

```
}{%
```

`\iso@printmonthday@american` Prints the month and the day given as two arguments (`{mm}{dd}`) in the current date format.

```
  \def\iso@printmonthday@american#1#2{%
```

Numeric and short date format: `mm/dd/`

```
    \ifthenelse{\equal{\iso@dateformat}{numeric}\or%
      \equal{\iso@dateformat}{short}}{%
      \printnumber[10]{#1}/\printnumber[10]{#2}/}{%
```

ISO date format: `mm-dd`

```
    \ifthenelse{\equal{\iso@dateformat}{iso}}{%
      \printnumber[10]{#1}\iso@isodash\printnumber[10]{#2}}{%
```

LaTeX date format: `mm/dd`

```
      \ifthenelse{\equal{\iso@dateformat}{TeX}}{%
        \printnumber[10]{#1}/\printnumber[10]{#2}}{%
```

Original date format: `mmm d`

```
        \ifthenelse{\equal{\iso@dateformat}{orig}}{%
          \begingroup%
          \edef\lmonth{#1}%
          \def\month{\lmonth}\month@english%
          \endgroup
          ~\printnumber{#2}%
          }{}}}}%
  }
```

`\iso@printdate@american` Prints the date given as three arguments (`{yyyy}{mm}{dd}`) in the actual date format.

```
\def\iso@printdate@american#1#2#3{%
```

ISO date format: `yyyy-\iso@printmonthday@american`

```
  \ifthenelse{\equal{\iso@dateformat}{iso}}{\number#1\iso@isodash}{%
```

LaTeX date format: `yyyy/\iso@printmonthday@american`

```
    \ifthenelse{\equal{\iso@dateformat}{TeX}}{\number#1/}{}}%
  \iso@printmonthday@american{\number#2}{\number#3}%
```

Numeric date format: `\iso@printmonthday@american yyyy`

```
  \ifthenelse{\equal{\iso@dateformat}{numeric}}{\number#1}{%
```

Original date format: `\iso@printmonthday@american,~yyyy`

```
    \ifthenelse{\equal{\iso@dateformat}{orig}}{,~\number#1}{%
```

Short date format: `\iso@printmonthday@american yy`

```
      \ifthenelse{\equal{\iso@dateformat}{short}}{%
        \iso@yeartwo{\number#1}}{}}}%
}
```

`\iso@printdate@USenglish` Just a second name for `\iso@printdate@UKamerican`.

```
\def\iso@printdate@USenglish{\iso@printdate@american}
```

`\iso@dateamerican` This command redefines the `\today` command to print in the actual date format.

```
\def\iso@dateamerican{%
  \def\today{\iso@printdate@american{\year}{\month}{\day}}}%
```

`\iso@daterange@american` This command takes six arguments (`{yyyy1}{mm1}{dd1}{yyyy2}{mm2}{dd2}`) and prints the corrosponding date range in the actual date format.

```
\def\iso@daterange@american#1#2#3#4#5#6{%
```

ISO or LaTeX date format.

```
\ifthenelse{\equal{\iso@dateformat}{iso}\or%
            \equal{\iso@dateformat}{TeX}}{%
```

Print the start date.

```
    \csname iso@printdate@\iso@languagename\endcsname{#1}{#2}{#3}%
    \iso@rangesign%
```

If year and month are equal, only print the day of the end date. If only the year is equal, only print month and day of the end date. Otherwise print the whole end date.

```
    \ifthenelse{\equal{\number#1}{\number#4}}{%
      \ifthenelse{\equal{\number#2}{\number#5}}{\printnumber[10]{#6}%
        }{\iso@printmonthday@american{#5}{#6}}}{%
      \csname iso@printdate@\iso@languagename\endcsname{%
        #4}{#5}{#6}}}{%
```

Original date format.

If year and month are equal, print `mmm d1 to d2, yyyy`. If only the year is equal, print `mmm1 d1 to mmm2 d2, yyyy`. Otherwise print the whole start and end date.

```
    \ifthenelse{\equal{\iso@dateformat}{orig}}{%
      \ifthenelse{\equal{\number#1}{\number#4}}{%
        \ifthenelse{\equal{\number#2}{\number#5}}{%
          \iso@printmonthday@american{#2}{#3}\iso@rangesign%
          \printnumber{#6},~\number#4}{%
          \iso@printmonthday@american{#2}{#3}\iso@rangesign%
          \csname iso@printdate@\iso@languagename\endcsname{%
            #4}{#5}{#6}}}{%
        \csname iso@printdate@\iso@languagename\endcsname{#1}{#2}{#3}%
        \iso@rangesign%
        \csname iso@printdate@\iso@languagename\endcsname{%
          #4}{#5}{#6}}}{%
```

Numeric or short date format.

If year and month are equal, only print the day of the end date. Otherwise print the whole end date.

```
      \ifthenelse{\equal{\number#1}{\number#4}}{%
      \iso@printmonthday@american{#2}{#3}}{%
      \csname iso@printdate@\iso@languagename\endcsname{#1}{#2}{#3}}%
```

Print the end date.

```
    \iso@rangesign\csname iso@printdate@\iso@languagename\endcsname{%
      #4}{#5}{#6}}%
  }%
}
```

`\iso@rangesign@american` Set the word between start and end date in a date range to " to ".

```
\def\iso@rangesign@american{~to~}
```

Define the language name that will the active language for isodate if none of the packages babel.sty, german.sty, and ngerman.sty is loaded and if this is the last language that is used for isodate. If one of the above packages is used this definition will be overridden by the command `\languagename` that will always return the current used language.

```
\def\iso@languagename{american}%
```

The end of the American section.

```
}
```

Redefine the command dateLanguage that is used by babel.sty, german.sty, and ngerman.sty to switch to the original English/American date format to enable the use of different date formats.

Do this only if \iso@dateLanguage is defined.

```
\AtBeginDocument{%
  \ifx\undefined\iso@dateenglish\else
    \def\dateenglish{\iso@dateenglish}%
    \def\dateUKenglish{\iso@dateenglish}%
  \fi
  \ifx\undefined\iso@dateamerican\else
    \def\dateamerican{\iso@dateamerican}%
    \def\dateUSenglish{\iso@dateamerican}%
  \fi
}
```

The end of the file.

E.4 Language definition file french.idf

Heading of the file:

```
\ProvidesFile{french.idf}[\filedate\space v\fileversion\space
  French support from the isodate package]
```

\iso@languageloaded

Define the command \iso@languageloaded in order to enable isodate.sty to determine if at least one language is loaded.

```
\let\iso@languageloaded\active
```

```
\def\month@french{\ifcase\month\or
  janvier\or f\'evrier\or mars\or avril\or mai\or juin\or
  juillet\or ao\^ut\or septembre\or octobre\or novembre\or
  d\'ecembre\fi}
```

```
  \def\iso@printmonthday@french#1#2{%
    \ifthenelse{\equal{\iso@dateformat}{numeric}\or%
      \equal{\iso@dateformat}{short}}{%
      \printnumber[10]{#2}/\printnumber[10]{#1}/}{%
      \ifthenelse{\equal{\iso@dateformat}{iso}}{%
        \iso@isodash\printnumber[10]{#1}\iso@isodash%
        \printnumber[10]{#2}}{%
        \ifthenelse{\equal{\iso@dateformat}{TeX}}{%
          /\printnumber[10]{#1}/\printnumber[10]{#2}}{%
          \ifthenelse{\equal{\iso@dateformat}{orig}}{%
            \begingroup
            \edef\lday{#2}\edef\day{\lday}%
            \edef\lmonth{#1}\def\month{\lmonth}%
            \number\day\ifnum1=\day \noexpand\ier\fi~\month@french%
            \endgroup
            }{}}}}%
    }
```

```
  \def\iso@printdate@french#1#2#3{%
    \ifthenelse{\equal{\iso@dateformat}{iso}\or%
      \equal{\iso@dateformat}{TeX}}{\number#1}{}%
    \iso@printmonthday@french{\number#2}{\number#3}%
    \ifthenelse{\equal{\iso@dateformat}{numeric}}{\number#1}{%
```

```
   \ifthenelse{\equal{\iso@dateformat}{orig}}{~\number#1}{%
    \ifthenelse{\equal{\iso@dateformat}{short}}{%
     \iso@yeartwo{\number#1}}{}}}%
  }
 \def\iso@datefrench{%
  \def\today{\iso@printdate@french{\year}{\month}{\day}}}%
\def\iso@daterange@french#1#2#3#4#5#6{%
 \ifthenelse{\equal{\iso@dateformat}{iso}\or%
       \equal{\iso@dateformat}{TeX}}{%
  \csname iso@printdate@\iso@languagename\endcsname{#1}{#2}{#3}%
  \iso@rangesign%
  \ifthenelse{\equal{\number#1}{\number#4}}{%
   \ifthenelse{\equal{\number#2}{\number#5}}{\printnumber[10]{#6}%
    }{\iso@printmonthday@french{#5}{#6}}}{%
   \csname iso@printdate@\iso@languagename\endcsname{#4}{#5}{#6}}}{%
  \ifthenelse{\equal{\number#1}{\number#4}}{%
   \ifthenelse{\equal{\number#2}{\number#5}}{%
    \ifthenelse{\equal{\iso@dateformat}{orig}}{%
     \begingroup
      \edef\lday{#3}\edef\day{\lday}%
      \number\day\ifnum1=\day \noexpand\ier\fi
      \endgroup}{\printnumber[10]{#3}}%
    }{\iso@printmonthday@french{#2}{#3}}}{%
   \csname iso@printdate@\iso@languagename\endcsname{#1}{#2}{#3}}%
  \iso@rangesign\csname iso@printdate@\iso@languagename\endcsname{%
   #4}{#5}{#6}%
}{%
  }%
}
\def\iso@rangesign@french{~--~}
```

Define the language name that will the active language for isodate if none of the packages babel.sty, german.sty, and ngerman.sty is loaded and if this is the last language that is used for isodate. If one of the above packages is used this definition will be overridden by the command `\languagename` that will always return the current used language.

```
\def\iso@languagename{french}%
```

`\datefrenchb` has to be defined additionally because babel starts with language frenchb instead of french.

```
\AtBeginDocument{%
 \ifx\undefined\iso@datefrench\else
  \def\datefrench{\iso@datefrench}%
  \def\datefrenchb{\iso@datefrench}%
 \fi
}
```

The end of the file.

E.5 Language definition file german.idf

Heading of the file:

```
\ProvidesFile{german.idf}[\filedate\space v\fileversion\space
 German support from the isodate package]
```

`\iso@languageloaded` Define the command `\iso@languageloaded` in order to enable `isodate.sty` to determine if at least one language is loaded.

```
\let\iso@languageloaded\active
```

Define spaces between day and month resp. month and year. `dm` stands for day-month and `my` for month-year. The defaults are taken from the Duden [2].

```
\def\iso@dmsepgerman{\,}%
\def\iso@mylongsepgerman{~}%
\def\iso@myshortsepgerman{\,}%
```

`\daymonthsepgerman` Change space between day and month in numeric date formats for the German language. The only parameter is the new spacing.

```
\DeclareRobustCommand*{\daymonthsepgerman}[1]{\def\iso@dmsepgerman{#1}}
%    \begin{macrocode}
% \end{macro}
% \begin{macro}{\monthyearsepgerman}
% Change space between month and year in numeric date formats for the
% German language. The first parameter is the new spacing for the long
% format and the second for the short format.
%    \begin{macrocode}
\DeclareRobustCommand*{\monthyearsepgerman}[2]{%
  \def\iso@mylongsepgerman{#1}%
  \def\iso@myshortsepgerman{#2}}
```

```
\def\month@german{\ifcase\month\or
  Januar\or Februar\or M\"arz\or April\or Mai\or Juni\or
  Juli\or August\or September\or Oktober\or November\or Dezember\fi}
\def\month@ngerman{\month@german}
\def\month@austrian{\ifnum1=\month
  J\"anner\else \month@german\fi}
\def\month@naustrian{\month@austrian}
```

```
\@namedef{iso@printmonthday@\CurrentOption}#1#2{%
  \ifthenelse{\equal{\iso@dateformat}{numeric}\or%
              \equal{\iso@dateformat}{short}}{%
    \printnumber[10]{#2}.\iso@dmsepgerman\printnumber[10]{#1}.}{%
    \ifthenelse{\equal{\iso@dateformat}{iso}}{%
      \printnumber[10]{#1}\iso@isodash\printnumber[10]{#2}}{%
      \ifthenelse{\equal{\iso@dateformat}{TeX}}{%
        \printnumber[10]{#1}/\printnumber[10]{#2}}{%
        \ifthenelse{\equal{\iso@dateformat}{orig}}{%
          \printnumber{#2}.~\begingroup
          \edef\lmonth{#1}%
          \def\month{\lmonth}\csname month@\iso@languagename\endcsname%
          \endgroup
          }{}}}}%
}
```

```
\@namedef{iso@printdate@\CurrentOption}#1#2#3{%
  \ifthenelse{\equal{\iso@dateformat}{iso}}{\number#1\iso@isodash}{%
    \ifthenelse{\equal{\iso@dateformat}{TeX}}{\number#1/}{}}%
  \csname iso@printmonthday@\iso@languagename\endcsname{%
    \number#2}{\number#3}%
  \ifthenelse{\equal{\iso@dateformat}{numeric}}{%
```

```
    \iso@mylongsepgerman\number#1}{%
    \ifthenelse{\equal{\iso@dateformat}{orig}}{~\number#1}{%
      \ifthenelse{\equal{\iso@dateformat}{short}}{%
        \iso@myshortsepgerman\iso@yeartwo{\number#1}}{}}}%
}
\@namedef{iso@daterange@\CurrentOption}#1#2#3#4#5#6{%
  \ifthenelse{\equal{\iso@dateformat}{iso}\or%
              \equal{\iso@dateformat}{TeX}}{%
    \csname iso@printdate@\iso@languagename\endcsname{#1}{#2}{#3}%
    \iso@rangesign%
    \ifthenelse{\equal{\number#1}{\number#4}}{%
      \ifthenelse{\equal{\number#2}{\number#5}}{\printnumber[10]{#6}%
        }{\csname iso@printmonthday@\iso@languagename\endcsname{#5}{#6}}}{%
      \csname iso@printdate@\iso@languagename\endcsname{#4}{#5}{#6}}}{%
    \ifthenelse{\equal{\number#1}{\number#4}}{%
      \ifthenelse{\equal{\number#2}{\number#5}}{%
        \ifthenelse{\equal{\iso@dateformat}{orig}}{%
          \printnumber{#3}}{\printnumber[10]{#3}}.%
        }{\csname iso@printmonthday@\iso@languagename\endcsname{%
          #2}{#3}}}{%
      \csname iso@printdate@\iso@languagename\endcsname{#1}{#2}{#3}}%
    \iso@rangesign\csname iso@printdate@\iso@languagename\endcsname{%
      #4}{#5}{#6}%
  }%
}
\@namedef{iso@rangesign@\CurrentOption}{~bis~}
\ifthenelse{\equal{\CurrentOption}{german}}{%
  \def\iso@dategerman{%
    \def\today{\iso@printdate@german{\year}{\month}{\day}}}%
```

Define the language name that will the active language for isodate if none of the packages babel.sty, german.sty, and ngerman.sty is loaded and if this is the last language that is used for isodate. If one of the above packages is used this definition will be overridden by the command `\languagename` that will always return the current used language.

```
  \def\iso@languagename{german}%
}{%
\ifthenelse{\equal{\CurrentOption}{ngerman}}{%
  \def\iso@datengerman{%
    \def\today{\iso@printdate@ngerman{\year}{\month}{\day}}}%
```

Define the language name that will the active language for isodate if none of the packages babel.sty, german.sty, and ngerman.sty is loaded and if this is the last language that is used for isodate. If one of the above packages is used this definition will be overridden by the command `\languagename` that will always return the current used language.

```
  \def\iso@languagename{ngerman}%
}{%
\ifthenelse{\equal{\CurrentOption}{austrian}}{%
  \def\iso@dateaustrian{%
    \def\today{\iso@printdate@austrian{\year}{\month}{\day}}}%
```

Define the language name that will the active language for isodate if none of the packages babel.sty, german.sty, and ngerman.sty is loaded and if this is the last language that is used for isodate. If one of the above packages is used this definition will be overridden by the command `\languagename` that will always return the current used language.

```
  \def\iso@languagename{austrian}%
}{%
```

```
\ifthenelse{\equal{\CurrentOption}{naustrian}}{%
  \def\iso@datenaustrian{%
    \def\today{\iso@printdate@naustrian{\year}{\month}{\day}}}%
```

Define the language name that will the active language for isodate if none of the packages babel.sty, german.sty, and ngerman.sty is loaded and if this is the last language that is used for isodate. If one of the above packages is used this definition will be overridden by the command `\languagename` that will always return the current used language.

```
  \def\iso@languagename{naustrian}%
}{%
}}}}
```

```
\AtBeginDocument{%
  \ifx\undefined\iso@dategerman\else
    \def\dategerman{\iso@dategerman}%
  \fi
  \ifx\undefined\iso@datengerman\else
    \def\datengerman{\iso@datengerman}%
  \fi
  \ifx\undefined\iso@dateaustrian\else
    \def\dateaustrian{\iso@dateaustrian}%
  \fi
  \ifx\undefined\iso@datenaustrian\else
    \def\datenaustrian{\iso@datenaustrian}%
  \fi
}
```

The end of the file.

E.6 Language definition file norsk.idf

This file was provided by Svend Tollak Munkejord (svend.t.munkejord@energy.sintef.no).

Heading of the file:

```
%% Endret til norsk, 2001-04-30, STM
\ProvidesFile{norsk.idf}[\filedate\space v\fileversion\space
  Norwegian support from the isodate package]
```

`\iso@languageloaded` Define the command `\iso@languageloaded` in order to enable `isodate.sty` to determine if at least one language is loaded.

```
\let\iso@languageloaded\active
```

`\month@norsk` Prints the name of today's month in the long form for the original date format.

```
\def\month@norsk{\ifcase\month\or
    januar\or februar\or mars\or april\or mai\or juni\or
    juli\or august\or september\or oktober\or november\or desember\fi}
```

\iso@printmonthday@norsk Prints the month and the day given as two arguments ({mm}{dd}) in the current date format.

```
\def\iso@printmonthday@norsk#1#2{%
```

Numeric and short date format: dd/mm/

```
  \ifthenelse{\equal{\iso@dateformat}{numeric}\or%
    \equal{\iso@dateformat}{short}}{%
    \printnumber[10]{#2}/\printnumber[10]{#1}/}{%
```

ISO date format: -mm-dd

```
    \ifthenelse{\equal{\iso@dateformat}{iso}}{%
      \iso@isodash\printnumber[10]{#1}\iso@isodash\printnumber[10]{#2}}{%
```

LaTeX date format: /mm/dd

```
      \ifthenelse{\equal{\iso@dateformat}{TeX}}{%
        /\printnumber[10]{#1}/\printnumber[10]{#2}}{%
```

Original date format: d. mmm

```
        \ifthenelse{\equal{\iso@dateformat}{orig}}{%
          \printnumber{#2}.~\begingroup
          \edef\lmonth{#1}\def\month{\lmonth}%
          \month@norsk%
          \endgroup
          }{}}}}%
  }
```

\iso@printdate@norsk Prints the date given as three arguments ({yyyy}{mm}{dd}) in the actual date format

```
\def\iso@printdate@norsk#1#2#3{%
```

ISO or LaTeXdate format: yyyy\iso@printmonthday@norsk

```
  \ifthenelse{\equal{\iso@dateformat}{iso}\or%
    \equal{\iso@dateformat}{TeX}}{\number#1}{}%
  \iso@printmonthday@norsk{\number#2}{\number#3}%
```

numeric date format: \iso@printmonthday@norsk yyyy

```
  \ifthenelse{\equal{\iso@dateformat}{numeric}}{\number#1}{%
```

original date format: \iso@printmonthday@norsk~yyyy

```
    \ifthenelse{\equal{\iso@dateformat}{orig}}{~\number#1}{%
```

short date format: \iso@printmonthday@norsk yy

```
      \ifthenelse{\equal{\iso@dateformat}{short}}{%
        \iso@yeartwo{\number#1}}{%
        }}}%
  }
```

\iso@datenorsk This command redefines the \today command to print in the actual date format.

```
\def\iso@datenorsk{%
  \def\today{\iso@printdate@norsk{\year}{\month}{\day}}}%
```

\iso@daterange@norsk This command takes six arguments ({yyyy1}{mm1}{dd1}{yyyy2}{mm2}{dd2}) and prints the corrosponding date range in the actual date format.

```
\def\iso@daterange@norsk#1#2#3#4#5#6{%
```

ISO or LaTeX date format.

```
  \ifthenelse{\equal{\iso@dateformat}{iso}\or%
              \equal{\iso@dateformat}{TeX}}{%
```

Print the start date.

```
    \csname iso@printdate@\iso@languagename\endcsname{#1}{#2}{#3}%
    \iso@rangesign%
```

If year and month are equal, only print the day of the end date. If only the year is equal, only print month and day of the end date. Otherwise print the whole end date.

```
    \ifthenelse{\equal{\number#1}{\number#4}}{%
      \ifthenelse{\equal{\number#2}{\number#5}}{\printnumber[10]{#6}%
        }{\iso@printmonthday@norsk{#5}{#6}}}{%
      \csname iso@printdate@\iso@languagename\endcsname{#4}{#5}{#6}}}{%
```

Numeric, short, or original date format.

If year and month are equal, only print the day of the start date. If only the year is equal, only print month and day of the start date. Otherwise print the whole start date.

```
    \ifthenelse{\equal{\number#1}{\number#4}}{%
      \ifthenelse{\equal{\number#2}{\number#5}}{%
        \ifthenelse{\equal{\iso@dateformat}{orig}}{%
          \printnumber{#3}.}{\printnumber[10]{#3}}%
        }{\iso@printmonthday@norsk{#2}{#3}}}{%
      \csname iso@printdate@\iso@languagename\endcsname{#1}{#2}{#3}}%
```

Print the end date.

```
    \iso@rangesign\csname iso@printdate@\iso@languagename\endcsname{%
      #4}{#5}{#6}%
}{%
    }%
}
```

`\iso@rangesign@norsk` Sets the word between start and end date in a date range to “ til ”.

```
\def\iso@rangesign@norsk{~til~}
```

Define the language name that will the active language for isodate if none of the packages babel.sty, german.sty, and ngerman.sty is loaded and if this is the last language that is used for isodate. If one of the above packages is used this definition will be overridden by the command `\languagename` that will always return the current used language.

```
\def\iso@languagename{norsk}%
```

Redefine the command `\datenorsk` that is used by babel to switch to the original Norsk date format to enable the use of different date formats.

```
\AtBeginDocument{%
  \ifx\undefined\iso@datenorsk\else
    \def\datenorsk{\iso@datenorsk}%
  \fi
}
```

The end of the file.

Change History

2.00

General: Total reimplementation of the package. The old package has renamed to isodateo. 1

2.01

General: For the case that none of the packages babel, german, and ngerman is loaded there is a new macro `\iso@languagename` that contains the name of the last loaded language. If one of the packages is loaded it contains the current language. . . . 1

Handle case of not loaded language package babel, german and ngerman 12

2.02

General: Added Norwegian language by Svend Tollak Munkejord 24

Changed the umlauts to normal TEX commands to be able to use German dates without german.sty or babel.sty. 22

2.03

General: Allow change of spaces for German language 4, 22

Fixed a bug in the French language that caused not to switch to it correctly on startup. . . . 21

2.04

General: Added section for solvable problems. 6

Index

Numbers written in italic refer to the page where the corresponding entry is described, the ones underlined to the code line of the definition, the rest to the code lines where the entry is used.