Full credit is given to the above companies including the OS that this PDF file was generated!

Red Hat Enterprise Linux Release 9.2 Manual Pages on 'date.1p' command

$ man date.1p

DATE(1P) POSIX Programmer's Manual DATE(1P)

PROLOG

This manual page is part of the POSIX Programmer's Manual. The Linux implementation of this interface may differ (consult the corresponding Linux manual page for details of Linux behavior), or the interface may not be implemented on Linux.

NAME

date ? write the date and time

SYNOPSIS

date [-u] [+format]

date [-u] mmddhhmm[[cc]yy]

DESCRIPTION

The date utility shall write the date and time to standard output or attempt to set the system date and time. By default, the current date and time shall be written. If an operand beginning with '+' is speci? fied, the output format of date shall be controlled by the conversion specifications and other text in the operand.

OPTIONS

The date utility shall conform to the Base Definitions volume of POSIX.1?2017, Section 12.2, Utility Syntax Guidelines.

The following option shall be supported:

-u Perform operations as if the TZ environment variable was set to the string "UTC0", or its equivalent historical value of

"GMT0". Otherwise, date shall use the timezone indicated by the TZ environment variable or the system default if that variable is unset or null.

OPERANDS

The following operands shall be supported:

+format When the format is specified, each conversion specifier shall be replaced in the standard output by its corresponding value. All other characters shall be copied to the output without change. The output shall always be terminated with a <newline>.

Conversion Specifications

%a Locale's abbreviated weekday name.

%A Locale's full weekday name.

%b Locale's abbreviated month name.

%B Locale's full month name.

%c Locale's appropriate date and time representation.

%C Century (a year divided by 100 and truncated to an integer) as a decimal number [00,99].

%d Day of the month as a decimal number [01,31].

%D Date in the format mm/dd/yy.

%e Day of the month as a decimal number [1,31] in a two-digit field with leading <space> character fill.

%h A synonym for %b.

%H Hour (24-hour clock) as a decimal number [00,23].

%I Hour (12-hour clock) as a decimal number [01,12].

%j Day of the year as a decimal number [001,366].

%m Month as a decimal number [01,12].

%M Minute as a decimal number [00,59].

%n A <newline>.

%p Locale's equivalent of either AM or PM.

%r 12-hour clock time [01,12] using the AM/PM notation; in the POSIX locale, this shall be equivalent to %I:%M:%S %p.

%S Seconds as a decimal number [00,60].

%t A <tab>.

%T 24-hour clock time [00,23] in the format HH:MM:SS.

%u Weekday as a decimal number [1,7] (1=Monday).

%U Week of the year (Sunday as the first day of the week) as a decimal number [00,53]. All days in a new year preceding the first Sunday shall be considered to be in week 0.

%V Week of the year (Monday as the first day of the week) as a decimal number [01,53]. If the week con? taining January 1 has four or more days in the new year, then it shall be considered week 1; otherwise, it shall be the last week of the previous year, and the next week shall be week 1.

%w Weekday as a decimal number [0,6] (0=Sunday).

%W Week of the year (Monday as the first day of the week) as a decimal number [00,53]. All days in a new year preceding the first Monday shall be considered to be in week 0.

%x Locale's appropriate date representation.

%X Locale's appropriate time representation.

%y Year within century [00,99].

%Y Year with century as a decimal number.

%Z Timezone name, or no characters if no timezone is de? terminable.

%% A <percent-sign> character.

See the Base Definitions volume of POSIX.1?2017, Section 7.3.5, LC_TIME for the conversion specifier values in the POSIX locale.

Modified Conversion Specifications

Some conversion specifiers can be modified by the E and O modifier characters to indicate a different format or specification as specified in the LC_TIME locale description (see the Base Definitions volume of

POSIX.1?2017, Section 7.3.5, LC_TIME). If the corresponding keyword (see era, era_year, era_d_fmt, and alt_digits in the Base Definitions volume of POSIX.1?2017, Section 7.3.5, LC_TIME) is not specified or not supported for the current locale, the unmodified conversion specifier value shall be used.

%Ec Locale's alternative appropriate date and time representation.

%EC The name of the base year (period) in the locale's alternative representation.

%Ex Locale's alternative date representation.

%EX Locale's alternative time representation.

%Ey Offset from %EC (year only) in the locale's alternative repre? sentation.

%EY Full alternative year representation.

%Od Day of month using the locale's alternative numeric symbols.

%Oe Day of month using the locale's alternative numeric symbols.

%OH Hour (24-hour clock) using the locale's alternative numeric symbols.

%OI Hour (12-hour clock) using the locale's alternative numeric symbols.

%Om Month using the locale's alternative numeric symbols.

%OM Minutes using the locale's alternative numeric symbols.

%OS Seconds using the locale's alternative numeric symbols.

%Ou Weekday as a number in the locale's alternative representation (Monday = 1).

%OU Week number of the year (Sunday as the first day of the week) using the locale's alternative numeric symbols.

%OV Week number of the year (Monday as the first day of the week, rules corresponding to %V), using the locale's alternative nu? meric symbols.

%Ow Weekday as a number in the locale's alternative representation (Sunday = 0).

%OW Week number of the year (Monday as the first day of the week) using the locale's alternative numeric symbols.

%Oy Year (offset from %C) in alternative representation.

mmddhhmm[[cc]yy]

Attempt to set the system date and time from the value given in the operand. This is only possible if the user has appro? priate privileges and the system permits the setting of the system date and time. The first mm is the month (number); dd is the day (number); hh is the hour (number, 24-hour system); the second mm is the minute (number); cc is the century and is the first two digits of the year (this is optional); yy is the last two digits of the year and is optional. If century is not specified, then values in the range [69,99] shall re? fer to years 1969 to 1999 inclusive, and values in the range [00,68] shall refer to years 2000 to 2068 inclusive. The cur? rent year is the default if yy is omitted.

Note: It is expected that in a future version of this standard the default century inferred from a 2-digit year will change. (This would apply to all commands accepting a 2-digit year as input.)

STDIN

Not used.

INPUT FILES

None.

ENVIRONMENT VARIABLES

The following environment variables shall affect the execution of date:

LANG Provide a default value for the internationalization vari? ables that are unset or null. (See the Base Definitions vol? ume of POSIX.1?2017, Section 8.2, Internationalization Vari? ables for the precedence of internationalization variables used to determine the values of locale categories.)

LC_ALL If set to a non-empty string value, override the values of all the other internationalization variables.

LC_CTYPE Determine the locale for the interpretation of sequences of bytes of text data as characters (for example, single-byte as

opposed to multi-byte characters in arguments).

LC_MESSAGES

Determine the locale that should be used to affect the format and contents of diagnostic messages written to standard er? ror.

LC_TIME Determine the format and contents of date and time strings written by date.

NLSPATH Determine the location of message catalogs for the processing of LC_MESSAGES.

TZ Determine the timezone in which the time and date are writ? ten, unless the -u option is specified. If the TZ variable is unset or null and -u is not specified, an unspecified system default timezone is used.

ASYNCHRONOUS EVENTS

Default.

STDOUT

When no formatting operand is specified, the output in the POSIX locale shall be equivalent to specifying:

```
date "+%a %b %e %H:%M:%S %Z %Y"
```

STDERR

The standard error shall be used only for diagnostic messages.

OUTPUT FILES

None.

EXTENDED DESCRIPTION

None.

EXIT STATUS

The following exit values shall be returned:

0 The date was written successfully.

>0 An error occurred.

CONSEQUENCES OF ERRORS

Default.

The following sections are informative.

APPLICATION USAGE

Conversion specifiers are of unspecified format when not in the POSIX locale. Some of them can contain <newline> characters in some locales, so it may be difficult to use the format shown in standard output for parsing the output of date in those locales.

The range of values for %S extends from 0 to 60 seconds to accommodate the occasional leap second.

Although certain of the conversion specifiers in the POSIX locale (such as the name of the month) are shown with initial capital letters, this need not be the case in other locales. Programs using these fields may need to adjust the capitalization if the output is going to be used at the beginning of a sentence.

The date string formatting capabilities are intended for use in Grego? rian-style calendars, possibly with a different starting year (or years). The %x and %c conversion specifications, however, are intended for local representation; these may be based on a different, non-Grego? rian calendar.

The %C conversion specification was introduced to allow a fallback for the %EC (alternative year format base year); it can be viewed as the base of the current subdivision in the Gregorian calendar. The century number is calculated as the year divided by 100 and truncated to an in? teger; it should not be confused with the use of ordinal numbers for centuries (for example, ``twenty-first century''.) Both the %Ey and %y can then be viewed as the offset from %EC and %C, respectively.

The E and O modifiers modify the traditional conversion specifiers, so that they can always be used, even if the implementation (or the cur? rent locale) does not support the modifier.

The E modifier supports alternative date formats, such as the Japanese Emperor's Era, as long as these are based on the Gregorian calendar system. Extending the E modifiers to other date elements may provide an implementation-defined extension capable of supporting other calendar systems, especially in combination with the O modifier.

The O modifier supports time and date formats using the locale's alter? native numerical symbols, such as Kanji or Hindi digits or ordinal num?

ber representation.

Non-European locales, whether they use Latin digits in computational items or not, often have local forms of the digits for use in date for? mats. This is not totally unknown even in Europe; a variant of dates uses Roman numerals for the months: the third day of September 1991 would be written as 3.IX.1991. In Japan, Kanji digits are regularly used for dates; in Arabic-speaking countries, Hindi digits are used.

The %d, %e, %H, %I, %m, %S, %U, %w, %W, and %y conversion specifica? tions always return the date and time field in Latin digits (that is, 0 to 9). The %O modifier was introduced to support the use for display purposes of non-Latin digits. In the LC_TIME category in localedef, the optional alt_digits keyword is intended for this purpose. As an exam? ple, assume the following (partial) localedef source:

```
alt_digits  "";"I";"II";"III";"IV";"V";"VI";"VII";"VIII" \
            "IX";"X";"XI";"XII"
d_fmt       "%e.%Om.%Y"
```

With the above date, the command:

```
date "+%x"
```

would yield 3.IX.1991. With the same d_fmt, but without the alt_digits, the command would yield 3.9.1991.

EXAMPLES

1. The following are input/output examples of date used at arbitrary times in the POSIX locale:

```
$ date
Tue Jun 26 09:58:10 PDT 1990
$ date "+DATE: %m/%d/%y%nTIME: %H:%M:%S"
DATE: 11/02/91
TIME: 13:36:16
$ date "+TIME: %r"
TIME: 01:36:32 PM
```

2. Examples for Denmark, where the default date and time format is %a %d %b %Y %T %Z:

```
$ LANG=da_DK.iso_8859-1 date
```

ons 02 okt 1991 15:03:32 CET

$ LANG=da_DK.iso_8859-1 \

date "+DATO: %A den %e. %B %Y%nKLOKKEN: %H:%M:%S"

DATO: onsdag den 2. oktober 1991

KLOKKEN: 15:03:56

3. Examples for Germany, where the default date and time format is %a %d.%h.%Y, %T %Z:

$ LANG=De_DE.88591 date

Mi 02.Okt.1991, 15:01:21 MEZ

$ LANG=De_DE.88591 date "+DATUM: %A, %d. %B %Y%nZEIT: %H:%M:%S"

DATUM: Mittwoch, 02. Oktober 1991

ZEIT: 15:02:02

4. Examples for France, where the default date and time format is %a %d %h %Y %Z %T:

$ LANG=Fr_FR.88591 date

Mer 02 oct 1991 MET 15:03:32

$ LANG=Fr_FR.88591 date "+JOUR: %A %d %B %Y%nHEURE: %H:%M:%S"

JOUR: Mercredi 02 octobre 1991

HEURE: 15:03:56

RATIONALE

Some of the new options for formatting are from the ISO C standard. The -u option was introduced to allow portable access to Coordinated Uni? versal Time (UTC). The string "GMT0" is allowed as an equivalent TZ value to be compatible with all of the systems using the BSD implemen? tation, where this option originated.

The %e format conversion specification (adopted from System V) was added because the ISO C standard conversion specifications did not pro? vide any way to produce the historical default date output during the first nine days of any month.

There are two varieties of day and week numbering supported (in addi? tion to any others created with the locale-dependent %E and %O modifier characters):

* The historical variety in which Sunday is the first day of the week

and the weekdays preceding the first Sunday of the year are consid? ered week 0. These are represented by %w and %U. A variant of this is %W, using Monday as the first day of the week, but still refer? ring to week 0. This view of the calendar was retained because so many historical applications depend on it and the ISO C standard strftime() function, on which many date implementations are based, was defined in this way.

* The international standard, based on the ISO 8601:2004 standard where Monday is the first weekday and the algorithm for the first week number is more complex: If the week (Monday to Sunday) con? taining January 1 has four or more days in the new year, then it is week 1; otherwise, it is week 53 of the previous year, and the next week is week 1. These are represented by the new conversion speci? fications %u and %V, added as a result of international comments.

FUTURE DIRECTIONS

None.

SEE ALSO

The Base Definitions volume of POSIX.1?2017, Section 7.3.5, LC_TIME, Chapter 8, Environment Variables, Section 12.2, Utility Syntax Guide? lines

The System Interfaces volume of POSIX.1?2017, fprintf(), strftime()

COPYRIGHT

files to man page format. To report such errors, see https://www.ker?

nel.org/doc/man-pages/reporting_bugs.html .