APPENDIX C

Date/Time Format Specfication

The date/time field parsing is supported using the Unix `strptime()` standard C library function.

The **strptime()** function is the converse function to **strftime()** and converts the character string pointed to by *s* to values which are stored in the *tm* structure pointed to by *tm*, using the format specified by *format*. Here ***format*** is a character string that consists of field descriptors and text characters, reminiscent of scanf(3). Each field descriptor consists of a % character followed by another character that specifies the replacement for the field descriptor. All other characters in the *format* string must have a matching character in the input string, except for whitespace, which matches zero or more whitespace characters in the input string.

The **strptime()** function processes the input string from left to right. Each of the three possible input elements (whitespace, literal, or format) are handled one after the other. If the input cannot be matched to the format string the function stops. The remainder of the format and input strings are not processed.

The supported input field descriptors are listed below. In case a text string (such as a weekday or month name) is to be matched, the comparison is case insensitive. In case a number is to be matched, leading zeros are permitted but not required.

%%

The **%** character.

%a or **%A**

The weekday name according to the current locale, in abbreviated form or the full name.

%b or **%B** or **%h**

The month name according to the current locale, in abbreviated form or the full name.

%c

The date and time representation for the current locale.

%C

The century number (0-99).

%d or **%e**

The day of month (1-31).

%D

Equivalent to %m/%d/%y. (This is the American style date, very confusing to non-Americans, especially since %d/%m/%y is widely used in Europe. The ISO 8601 standard format is %Y-%m-%d.)

%H

The hour (0-23).

%I

The hour on a 12-hour clock (1-12).

%j

The day number in the year (1-366).

%m

The month number (1-12).

%M

The minute (0-59).

%n or **%t**

Arbitrary whitespace.

%p

The locale's equivalent of AM or PM. (Note: there may be none.)

%r

The 12-hour clock time (using the locale's AM or PM). In the POSIX locale equivalent to %I:%M:%S %p. If *t_fmt_ampm* is empty in the LC_TIME part of the current locale then the behaviour is undefined.

%R

Equivalent to %H:%M.

%S

The second (0-60; 60 may occur for leap seconds; earlier also 61 was allowed).

%T

Equivalent to %H:%M:%S.

%U

The week number with Sunday the first day of the week (0-53). The first Sunday of January is the first day of week 1.

%w

The weekday number (0-6) with Sunday = 0.

%W

The week number with Monday the first day of the week (0-53). The first Monday of January is the first day of week 1.

%x

The date, using the locale's date format.

%X

The time, using the locale's time format.

%y

The year within century (0-99). When a century is not otherwise specified, values in the range 69-99 refer to years in the twentieth century (1969-1999); values in the range 00-68 refer to years in the twenty-first century (2000-2068).

%Y

The year, including century (for example, 1991).

Some field descriptors can be modified by the E or O modifier characters to indicate that an alternative format or specification should be used. If the alternative format or specification does not exist in the current locale, the unmodified field descriptor is used.

The E modifier specifies that the input string may contain alternative locale-dependent versions of the date and time representation:

%Ec

The locale's alternative date and time representation.

%EC

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

%Ex

The locale's alternative date representation.

%EX

The locale's alternative time representation.

%Ey

The offset from %EC (year only) in the locale's alternative representation.

%EY

The full alternative year representation.

The O modifier specifies that the numerical input may be in an alternative locale-dependent format:

%Od or **%Oe**

The day of the month using the locale's alternative numeric symbols; leading zeros are permitted but not required.

%OH

The hour (24-hour clock) using the locale's alternative numeric symbols.

%OI

The hour (12-hour clock) using the locale's alternative numeric symbols.

%Om

The month using the locale's alternative numeric symbols.

%OM

The minutes using the locale's alternative numeric symbols.

%OS

The seconds using the locale's alternative numeric symbols.

%OU

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

%Ow

The number of the weekday (Sunday=0) using the locale's alternative numeric symbols.

%OW

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

%Oy

The year (offset from %C) using the locale's alternative numeric symbols.

%F

Equivalent to **%Y-%m-%d**, the ISO 8601 date format.

%g

The year corresponding to the ISO week number, but without the century (0-99).

%G

The year corresponding to the ISO week number. (For example, 1991.)

%u

The day of the week as a decimal number (1-7, where Monday = 1).

%V

The ISO 8601:1988 week number as a decimal number (1-53). If the week (starting on Monday) containing 1 January has four or more days in the new year, then it is considered week 1. Otherwise, it is the last week of the previous year, and the next week is week 1.

%z

An RFC-822/ISO 8601 standard time zone specification.

%Z

The timezone name.

Similarly, because of GNU extensions to *strftime*, %k is accepted as a synonym for %H, and %l should be accepted as a synonym for %I, and %P is accepted as a synonym for %p. Finally

%s

The number of seconds since the epoch, i.e., since 1970-01-01 00:00:00 UTC. Leap seconds are not counted unless leap second support is available.