Document number: SC22/WG14 N658
(J11 97-021)

Title: POSIX Alignment

Author: Keld Simonsen
Author affiliation: DKUUG
Postal address: Fruebjergvej 3, DK-2100 København Ø
Email address: keld@dkuug.dk
Telephone number: +45 3917-9944
Fax number: +45 3325-6543
Sponsor: DS
Date: 1997-02-10
Proposal category:
    __ Editorial change/non-normative contribution
    XX Correction
    XX New feature
    __ Addition to obsolescent feature list
    __ Other (please specify)
Area of standard affected:
    XX Environment
    XX Language
    __ Preprocessor
    XX Library
       XX Macro/typedef/tag name
       XX Function
       XX Header
    __ Other (please specify)
Prior art: ISO/IEC 9945 POSIX standards
Target audience: general

Related documents: N431 (Rationale and analysis), N507, N538, N586
Proposal attached: proposal paper
Review committeee: Keld Simonsen, Clive Feather, Rex Jaeschke

Abstract:
The paper gives proposals for alignment of C9X with the POSIX standards wrt internationalization features.

Introduction

This paper gives proposals for changes to the C standard to align it with the POSIX standards POSIX System API (C language) (POSIX-1), and ISO/IEC 9945-2:1993 POSIX Shell and Utilities (POSIX-2). It does not cover newer proposals for POSIX or other related specifications, that are not yet international standards.

It builds on the document N431, which gave an overview of internationalisation in C and POSIX standards, a comparison of the functionality and features provided, and also mentioned other incompatibilities between C and POSIX standards. Thus N431 gave the background and rationale for the proposed changes, and it was decided in the Copenhagen meeting to do further work based on N431. The paper here descibes in detail what the changes should be.

Internationalization may be abbreviated as I18N in the following.

The following section numbers refer to the C9X Draft 8.

Changes to N586

Changes to the N586 document are:

LC_MESSAGES description with yesexpr and noexpr deleted
Reserving a "std/" namespace for setlocale() deleted
Deleted numerical thousands and decimal seperator being more than one character.

Changes in N586 (1996-06-26)

Changes to the N507/N538 document are:
iswblank() function added
%F added in strftime() to indicate YYYY-MM-DD
POSIX compatibility section added
Text for inclusion added

7.3.1 Character testing functions

POSIX-2 adds in its section 2.5.2 a class "blank", consisting initially of the characters <space> and <tab>. This character class should be added, possibly by adding a function isblank() that is similar to the isspace() function except that the test is for a standard blank character, and the characters covered initially only are space (' ') and horizontal tab ('\t'). Similary a function iswblank() should be added. Text for 7.3.1.12:

"7.3.1.12 the isblank function

synopsis

[1]

```
#include <ctype.h>
int isblank(int c);
```

Description:

[2] The isblank function tests for any charcacter that is a standard or locale-specific blank character. The standard blank characters are the following: space (' '), and horizontal tab ('\t').

7.3.2 Character case mapping functions

C has no statement on locale dependence, nor how the correspondence is defined. This can be described by adding: "as specified by the current locale" to both the toupper() and tolower() descriptions, so it reads (for tolower):

"If the argument is a character for which isupper is true and there is a corresponding character as specified by the current locale for which islower is true, the tolower function returns the corresponding character; otherwise, the argument is returned unchanged."

7.4 Localization

The POSIX-2 standard was approved after adoption of the C standard, and it contains a format for specifying locales and accompanying charmaps. This is a valuable and standardized way of specifying locales, on the other hand many C compilers do not operate under a POSIX operating system. It is proposed to add in 7.4 after the macro (LC_ALL etc) section in a non-normative note:

"Note:
POSIX-2 specifies locale and charmap formats that may be used to specify locales for C."

A reference to the POSIX-2 standard should be added to the informative bibliography.

7.4.2 Numeric formatting

In POSIX-2 all char variables may use -1 to indicate that the value is not available, instead of the value CHAR_MAX, so to be campatible, it is proposed that -1 is added to the values indicating this. Thus after "CHAR_MAX" add: "or -1". Also here change "can" to "may".

Note from Keld: would that not mean that the chars should be signed?

7.4.2.1 p_sign_posn and n_sign_posn

POSIX has added a 5th value and thus it is proposed to add:

"5 A space separates the symbol and the sign string, if adjacent."

A table giving the formats for the combinations of *_sign_posn needs to be present, this is not given here yet.

7.5.2.1 int_curr_symbol different from currency_symbol

p_sign_posn and n_sign_posn are also applicable to int_curr_symbol in POSIX-2.

As there may be differences between the order of how local currency is written and how international currency is written, it is proposed to add the 4 following members of the lconv struct:

int_p_cs_precedes
int_p_sep_by_space
int_n_cs_precedes

int_n_sep_by_space

with equivalent wording as "p_cs_precedes" etc, where "currency_symbol" is replaced with "int_curr_symbol" in 7.5.2.1[3].

In section 7.5.2.1 the examples need to be enhanced.

7.14.3.5 strftime

The date utility in POSIX-2 4.15 has all of the formats of C's strftime() plus more, all of which are proposed to be added to strftime:

%C is replaced by the century (a year divided by 100 and truncated to an integer) as a decimal number (00-99)
%D is replaced by the date in the format mm/dd/yy
%e is replaced by the day of the month as a decimal number (1-31 in a two-digit field with leading <space> fill)
%F is replaced by the date in the format YYYY-MM-DD (ISO 8601 format)
%h a synonym for %b
%n is replaced by a <newline> character
%r is replaced by the 12 h clock time (01-12) using the AM/PM notation; in the "C" locale, this shall be equivalent to "%I:%M:%S %p"
%t is replaced by a <tab> character
%T is replaced by the 24 h clock time (00-23) in the format HH:MM:SS.
%u is replaced by the 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 is replaced by the week of the year (Monday as the first day of the week) as a decimal number (00-53). The method for determining the week number shall be as specified in ISO 8601.

A number of modified field descriptors %O<d> and %E<d> are also defined (4.15.4.2). Some field descriptors 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. If the corresponding data (see era, era_year, era_d_fmt, and alt_digits) is not specified or not supported for the current locale, the unmodified field descriptor value shall be used.

Kelds note: I cannot see where era, era_year era_d_fmt and alt_digits are defined, as well as the normal date and time locale stuff. Should these not be added to 7.5[2] and 7.5.2.1[3]?

%Ec Locale's alternate date and time representation.
%EC The name of the base year (period) in the locale's alternate representation.
%Ex Locale's alternate date representation.
%Ey Offset from %EC (year only) in the locale's alternate representation.
%EY Full alternate year representation.

%Od Day of month using the locale's alternate numeric symbols.
%Oe Day of month using the locale's alternate numeric symbols.
%OH Hour (24-hour clock) using the locale's alternate numeric symbols.
%OI Hour (12-hour clock) using the locale's alternate numeric symbols.
%Om Month using the locale's alternate numeric symbols.
%OM Minutes using the locale's alternate numeric symbols.
%OS Seconds using the locale's alternate numeric symbols.
%OU Week number of the year (Sunday as the first day of the week) using the locale's alternate numeric symbols.
%Ow Weekday as number in the locale's alternate representation (Sunday=0).
%OW Week number of the year (Monday as the first day of the week) using the locale's alternate numeric symbols.
%Oy Year (offset from %C) in alternate representation.

POSIX-1

POSIX-1 defines the kernel interface, given in C language binding. For a lot of functionality, it does not define the I18N functionality, but relies on the C standard, which is included normatively with the C binding option of the standard. There is a separate section (8) in POSIX-1 giving the extensions defined by POSIX-1 in relation to the C standard.

The extensions cover the following functions: setlocale, rename, getenv, ctime, gmtime, localtime, mktime and strftime. Also fseek and exit are specified further.

Extensions to the time functions concern the use of the environment variable TZ, to override system defaults.

A number of operating system considerations is done for various C I/O functions.

POSIX-1 section 8 (12 pages) should be considered for technical corrigenda, or for inclusion in amendment/revision of the C standard.

Frank Farance has taken an action item on a proposal that addresses this.

Differences from POSIX

This proposal introduces the following changes from POSIX (all additions):

Adds to lconv struct:
int_p_cs_precedes
int_p-sep_by_space
int_n_cs_precedes
int_n_sep_by_space

Adds %F for ISO 8601 format in strftime()

Other specifications

An amendment (.2b) to POSIX-2 is currently underway (currently out for CD registration ballot) providing further specifications in the i18n area. This may be relevant to C functionality.

WG20 is working on an specification standards for cultural conventions, which has POSIX and C downwards compatible locale and charmap functionality. The extensions should be relevant for the C standard. This is currently WD stage, expected to go to CD registration Oct 1996.

WG21 is specifying a number of i18n and character functionality in their C++ standard. This is currently at CD stage.

It is proposed to watch these activities closely and align where possible.

<END OF DOCUMENT>