X Locale Database Specification

Yoshio Horiuchi, IBM Japan
X Locale Database Specification
by Yoshio Horiuchi
X Version 11, Release 7.7
Copyright © 1994 IBM Corporation

License to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of IBM not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

IBM DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS, AND NONINFRINGEMENT OF THIRD PARTY RIGHTS, IN NO EVENT SHALL IBM BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Copyright © 1994 X Consortium

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE X CONSORTIUM BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of the X Consortium shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from the X Consortium.

X Window System is a trademark of The Open Group.
# Table of Contents

1. LocaleDB ........................................................................................................... 1  
   General ........................................................................................................... 1  
   Database Format Definition ........................................................................... 1  
   Contents of Database ..................................................................................... 2  
   XLC_FONTSET Category ............................................................................... 2  
   XLC_XLOCALE Category ............................................................................... 3  
   Sample of X Locale Database ........................................................................ 5  
   Reference ....................................................................................................... 6
Chapter 1. LocaleDB

General

An X Locale Database contains the subset of a user's environment that depends on language, in X Window System. It is made up from one or more categories. Each category consists of some classes and sub-classes.

It is provided as a plain ASCII text file, so a user can change its contents easily. It allows a user to customize the behavior of internationalized portion of Xlib without changing Xlib itself.

This document describes;

• Database Format Definition
• Contents of Database in sample implementation

Since it is hard to define the set of required information for all platforms, only the flexible database format is defined. The available entries in database are implementation dependent.

Database Format Definition

The X Locale Database contains one or more category definitions. This section describes the format of each category definition.

The category definition consists of one or more class definitions. Each class definition has a pair of class name and class value, or has several subclasses which are enclosed by the left brace ({) and the right brace (}).

Comments can be placed by using the number sign character (#). Putting the number sign character on the top of the line indicates that the entire line is comment. Also, putting any whitespace character followed by the number sign character indicates that a part of the line (from the number sign to the end of the line) is comment. A line can be continued by placing backslash (\) character as the last character on the line; this continuation character will be discarded from the input. Comment lines cannot be continued on a subsequent line using an escaped new line character.

X Locale Database only accepts XPCS, the X Portable Character Set. The reserved symbols are; the quotation mark("), the number sign (#), the semicolon(;) the backslash(\), the left brace(}) and the right brace(}).

The format of category definition is;

| CategoryDefinition | ::= | CategoryHeader CategorySpec CategoryTrailer |
| CategoryHeader     | ::= | CategoryName NL |
| CategorySpec       | ::= | { ClassSpec } |
| CategoryTrailer    | ::= | "END" Delimiter CategoryName NL |
| CategoryName       | ::= | String |
| ClassSpec          | ::= | ClassName Delimiter ClassValue NL |
| ClassName          | ::= | String |
Contents of Database

The available categories and classes depend on implementation, because different platform will require different information set. For example, some platform have system locale but some platform don't. Furthermore, there might be a difference in functionality even if the platform has system locale.

In current sample implementation, categories listed below are available.

XLC_FONTSET: XFontSet relative information
XLC_XLOCALE: Character classification and conversion information

XLC_FONTSET Category

The XLC_FONTSET category defines the XFontSet relative information. It contains the CHARSET_REGISTRY-CHARSET_ENCODING name and character mapping side (GL, GR, etc), and is used in Output Method (OM).
LocaleDB

### class

<table>
<thead>
<tr>
<th>fsN</th>
<th>fsN</th>
<th>Nth fontset (N=0,1,2,...)</th>
</tr>
</thead>
<tbody>
<tr>
<td>charset</td>
<td>fsN</td>
<td>list of encoding name</td>
</tr>
<tr>
<td>font</td>
<td>fsN</td>
<td>list of font encoding name</td>
</tr>
</tbody>
</table>

fsN Includes an encoding information for Nth charset, where N is the index number (0,1,2,...). If there are 4 charsets available in current locale, 4 fontsets, fs0, fs1, fs2 and fs3, should be defined. This class has two subclasses, 'charset' and 'font'.

### charset

Specifies an encoding information to be used internally in Xlib for this fontset. The format of value is:

- EncodingInfo ::= EncodingName [ ":" EncodingSide ]
- EncodingName ::= CHARSET_REGISTRY-CHARSET_ENCODING
- EncodingSide ::= "GL" | "GR"

For detail definition of CHARSET_REGISTRY-CHARSET_ENCODING, refer to the X Logical Font Description Conventions document.

Example:

ISO8859-1:GL

### font

Specifies a list of encoding information which is used for searching appropriate font for this fontset. The left most entry has highest priority.

### XLC_XLOCALE Category

The XLC_XLOCALE category defines character classification, conversion and other character attributes.

<table>
<thead>
<tr>
<th>class</th>
<th>super class</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>encoding_name</td>
<td>encoding_name</td>
<td>codeset name</td>
</tr>
<tr>
<td>mb_cur_max</td>
<td>MB_CUR_MAX</td>
<td></td>
</tr>
<tr>
<td>state_depend_encoding</td>
<td>state dependent or not</td>
<td></td>
</tr>
<tr>
<td>wc_encoding_mask</td>
<td>for parsing wc string</td>
<td></td>
</tr>
<tr>
<td>wc_shift_bits</td>
<td>for conversion between wc and mb</td>
<td></td>
</tr>
<tr>
<td>csN</td>
<td>csN</td>
<td>Nth charset (N=0,1,2,...)</td>
</tr>
<tr>
<td>side</td>
<td>csN</td>
<td>mapping side (GL, etc)</td>
</tr>
<tr>
<td>length</td>
<td>csN</td>
<td>length of a character</td>
</tr>
<tr>
<td>mb_encoding</td>
<td>csN</td>
<td>for parsing mb string</td>
</tr>
<tr>
<td>wc_encoding</td>
<td>csN</td>
<td>for parsing wc string</td>
</tr>
<tr>
<td>ct_encoding</td>
<td>csN</td>
<td>list of encoding name for ct</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>-----------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>encoding_name</td>
<td>Specifies a codeset name of current locale.</td>
<td></td>
</tr>
<tr>
<td>mb_cur_max</td>
<td>Specifies a maximum allowable number of bytes in a multi-byte character. It is corresponding to MB_CUR_MAX of “ISO/IEC 9899:1990 C Language Standard”.</td>
<td></td>
</tr>
<tr>
<td>state_depend_encoding</td>
<td>Indicates a current locale is state dependent. The value should be specified “True” or “False”.</td>
<td></td>
</tr>
<tr>
<td>wc_encoding_mask</td>
<td>Specifies a bit-mask for parsing wide-char string. Each wide character is applied bit-and operation with this bit-mask, then is classified into the unique charset, by using ‘wc_encoding’.</td>
<td></td>
</tr>
<tr>
<td>wc_shift_bits</td>
<td>Specifies a number of bit to be shifted for converting from a multi-byte character to a wide character, and vice-versa.</td>
<td></td>
</tr>
<tr>
<td>csN</td>
<td>Includes a character set information for Nth charset, where N is the index number (0,1,2,...). If there are 4 charsets available in current locale, cs0, cs1, cs2 and cs3 should be defined. This class has five subclasses, ‘side’, ‘length’, ‘mb_encoding’ ‘wc_encoding’ and ‘ct_encoding’.</td>
<td></td>
</tr>
<tr>
<td>side</td>
<td>Specifies a mapping side of this charset. The format of this value is;</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Side ::= EncodingSide[&quot;:Default&quot;]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>The suffix &quot;:Default&quot; can be specified. It indicates that a character belongs to the specified side is mapped to this charset in initial state.</td>
<td></td>
</tr>
<tr>
<td>length</td>
<td>Specifies a number of bytes of a multi-byte character of this charset. It should not contain the length of any single-shift sequence.</td>
<td></td>
</tr>
<tr>
<td>mb_encoding</td>
<td>Specifies a list of shift sequence for parsing multi-byte string. The format of this value is;</td>
<td></td>
</tr>
<tr>
<td></td>
<td>MBEncoding ::= ShiftType ShiftSequence</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ShiftType ShiftSequence &quot;,&quot; MBEncoding</td>
</tr>
<tr>
<td>ShiftType</td>
<td>::= &quot;&lt;SS&gt;&quot;</td>
<td>&quot;&lt;LSL&gt;&quot;</td>
</tr>
<tr>
<td>ShiftSequence</td>
<td>::= SequenceValue</td>
<td>SequenceValue ShiftSequence</td>
</tr>
<tr>
<td>SequenceValue</td>
<td>::= NumericString</td>
<td></td>
</tr>
<tr>
<td>example:</td>
<td>&lt;LSL&gt; \x1b \x28 \x4a; &lt;LSL&gt; \x1b \x28 \x42</td>
<td></td>
</tr>
</tbody>
</table>
wc_encoding Specifies an integer value for parsing wide-character string. It is used to determine the charset for each wide character, after applying bit-and operation using 'wc_encoding_mask'. This value should be unique in all csN classes.

c_t_encoding Specifies a list of encoding information that can be used for Compound Text.

Sample of X Locale Database

The following is sample X Locale Database file.

#  XLocale Database Sample for ja_JP.euc
#
#
#      XLC_FONTSET category
#
XLC_FONTSET
#      fs0 class (7 bit ASCII)
fs0     {
   charset              ISO8859-1:GL
   font                 ISO8859-1:GL; JISX0201.1976-0:GL
}
#      fs1 class (Kanji)
fs1     {
   charset              JISX0208.1983-0:GL
   font                 JISX0208.1983-0:GL
}
#      fs2 class (Half Kana)
fs2     {
   charset              JISX0201.1976-0:GR
   font                 JISX0201.1976-0:GR
}
#      fs3 class (User Defined Character)
# fs3     {
#       charset             JISX0212.1990-0:GL
#       font                JISX0212.1990-0:GL
# } END XLC_FONTSET

#
#
#      XLC_XLOCALE category
#
XLC_XLOCALE
encoding_name             ja.euc
mb_cur_max                3
state_depend_encoding     False

wc_encoding_mask          \x000008080
wc_shift_bits             8
# cs0 class

```c
    cs0 {
        side           GL:Default
        length         1
        wc_encoding    \x00000000
        ct_encoding    ISO8859-1:GL; JISX0201.1976-0:GL
    }
```

# cs1 class

```c
    cs1 {
        side           GR:Default
        length         2
        wc_encoding    \x00008080
    }
```

# cs2 class

```c
    cs2 {
        side           GR
        length         1
        mb_encoding    <SS> \x8e
        wc_encoding    \x00000080
        ct_encoding    JISX0201.1976-0:GR
    }
```

# cs3 class

```c
    # cs3 {
    #     side           GL
    #     length         2
    #     mb_encoding    <SS> \x8f
    # #if HasWChar32
    #     wc_encoding    \x20000000
    # #else
    #     wc_encoding    \x00008000
    # #endif
    #     ct_encoding    JISX0212.1990-0:GL; JISX0212.1990-0:GR
    # }
```

END XLC_XLOCALE

Reference

[2] X Logical Font Description Conventions