Introducing Internationalized XView

3

XView is a user-interface toolkit based on the X Window System and the OPEN LOOK graphical user interface. It is included in the current Solaris release.

XView allows you to internationalize your OPEN LOOK applications by supporting Asian and many western European languages:

- Western European (ISO Latin-1) languages and Asian languages are supported with locale settings, localized text handling, and customized object layout.
- Asian languages have multibyte, wide characters, input method, and font set support.

You do not have to redesign or recompile internationalized applications. The goal is to have a single application binary operate in any of the supported locales.

In an internationalized XView application, language-specific application data (message strings, labels, and so on) is separate from the rest of the application. To localize the application (that is, to adapt the application to support a specific language) you need to modify only the language-specific data. Thus, the task of porting an internationalized application consists of, among other things, translating application-specific strings and modifying object layout.

The information in this manual builds on the contents of several other manuals published by SunSoft and O'Reilly and Associates. Check the preface to make sure you have the manuals you need.

3

Internationalization Features

The internationalization of XView applications is outlined in the *XView Programming Manual, Version 3,* published by O'Reilly and Associates. Specifically, XView supports the following internationalization features described in that programming manual:

- *Locale setting*
  Before running an internationalized application, users must select what language to run in. The locale setting feature allows the user to choose the language or cultural environment.
- *Localized text handling*
  As a developer, you need to be able to write application strings—error messages, menu labels, button labels—in the native language, and have those strings retrieved in the language specified by the locale. This process is called localized text handling.
- *Object layout*
  When an application is run in a non-native language, the layout of various objects may change. For example, the dimensions of objects containing strings, such as buttons and panels, may change. Object layout is the mechanism by which the screen location of objects is modified (depending on the display language) to accommodate these kinds of changes.

This manual describes additional internationalization features that are *not* documented in the *XView Programming Manual.*

- Wide character and multibyte characters
- Input method
- Font sets

Wide Characters and Multibyte Characters

English language applications use ASCII encoding to represent characters. Each character is encoded using one byte (actually 7 out of the 8 bits). Other languages have multiple character sets that sometimes contain extremely large numbers of characters. These languages require more than one byte to represent each character and must be encoded differently. The current release of XView uses Extended UNIX Code (EUC) encoding.

Certain XView attributes and functions have been modified to handle EUC multibyte characters. There are also wide character attributes and functions. Wide character attributes are suffixed with `_WC` (wide character) or `_WCS` (wide character string). Similarly, wide character functions are suffixed with `_wc` or `_wcs`. See Chapter 4, "Character Encoding for more information.

Input Method

Input method refers to how users enter text in an application. For example, to enter data in a typical European language application—say German—users simply type the information. Many Asian languages, however, consist of multiple character sets (for example, Japanese has two phonetic alphabets and one ideographic character set). These multiple character sets can consist of many thousands of characters and contain numerous homonyms for any particular word. Entering data in these languages requires special input handling. See Chapter 5, "Input Method for more information.

Font Sets

Most western European languages consist of a single character set, and only one font is necessary to support the language. Languages with multiple character sets require multiple fonts, which are grouped into font sets. The font handling API has been extended in the current XView release to handle font sets. See "Fonts" on page 45 for more information.

Compiling XView Programs

Use the following command line to compile your XView application:

```
cc -DOW_I18N -I$OPENWINHOME/include file.c
-L$OPENWINHOME/lib -lxview -lolgx -lX11 {-lintl -lw}
```

`-DOW_I18N=` Enables internationalization support

`-lxview=` XView library

`-lolgx=` OPEN LOOK graphics library

`-lX11=` X11 Release 5 library

The following optional flags link in libraries that may be needed by the application:

`-lintl`=Message cataloguing library. Needed if application uses `gettext` family of functions.

`-lw=` Wide character support library. Needed if the application uses wide character functions such as `getwchar()`.