Installation Issues 2

This chapter presents information on installation and system administration issues for the XGL library.

About the XGL Directory Structure

The XGL runtime library, font files, and error message files are installed with the Solaris product, and the XGL `include` files and documentation are installed with the Solaris Software Developer Kit (SDK). The Solaris product includes the XGL packages listed in Table 2-1.

Table 2-1 XGL Runtime Packages

Package Name	Package Contents
`SUNWxglrt`	XGL runtime library and the Roman_M stroke font
`SUNWxgler`	Localized error message files for `SUNWxglrt`
`SUNWxgldg`	Loadable pipeline libraries for generic drivers with specific error message files when applicable. The generic drivers are cg6 (GX, GXplus, TurboGX, and TurboGXplus), cfb (non-accelerated color frame buffers such as cg3 and cg8), mem (memory), xpex (combined xlib and pexlib), swp (software pipeline), and cgm (CGM Stream device) . XGL loadable libraries for other devices are included in the packages distributed with each device.

Table 2-1 XGL Runtime Packages *(Continued)*

Package Name	Package Contents
`SUNWxglft`	XGL stroke fonts
`SUNWxglh`	Copy of XGL `include` files installed into `/usr/openwin/share/include`

The SDK provides the XGL packages listed in Table 2-2. Note that the package `SUNWxglh`, which was formerly part of the SDK, is now delivered with the runtime environment and contains only XGL header files. The new SDK package `SUNWxglsk`, contains the Software Developer Kit files, including the header files, example programs, and man pages.

Table 2-2 XGL SDK Packages

Package Name	Package Contents
`SUNWxglsk`	XGL `include` files, example programs, and man pages.
`SUNWAxg`	AnswerBook versions of XGL documentation.

By default, the XGL runtime software packages are installed in `/opt/SUNWits/Graphics-sw/xgl`, the XGL SDK files are installed in `/opt/SUNWsdk/sdk_2.5/xgl`, and the AnswerBook software in `/opt/SUNWAxg`.

The `libxgl.so` file in the `lib` directory is a link to the `libxgl.so.3` file. The link is required by the System V Release 4 UNIX®runtime loader, since the loader does not recognize files that end in a release number.

Note – Previous symbolic links that were in the `/lib/pipelines` directory have been removed because they were unused.

The XGL runtime directory structure is shown in Figure 2-1 on page 17. The XGL SDK directory structure is shown in Figure 2-2 on page 18.

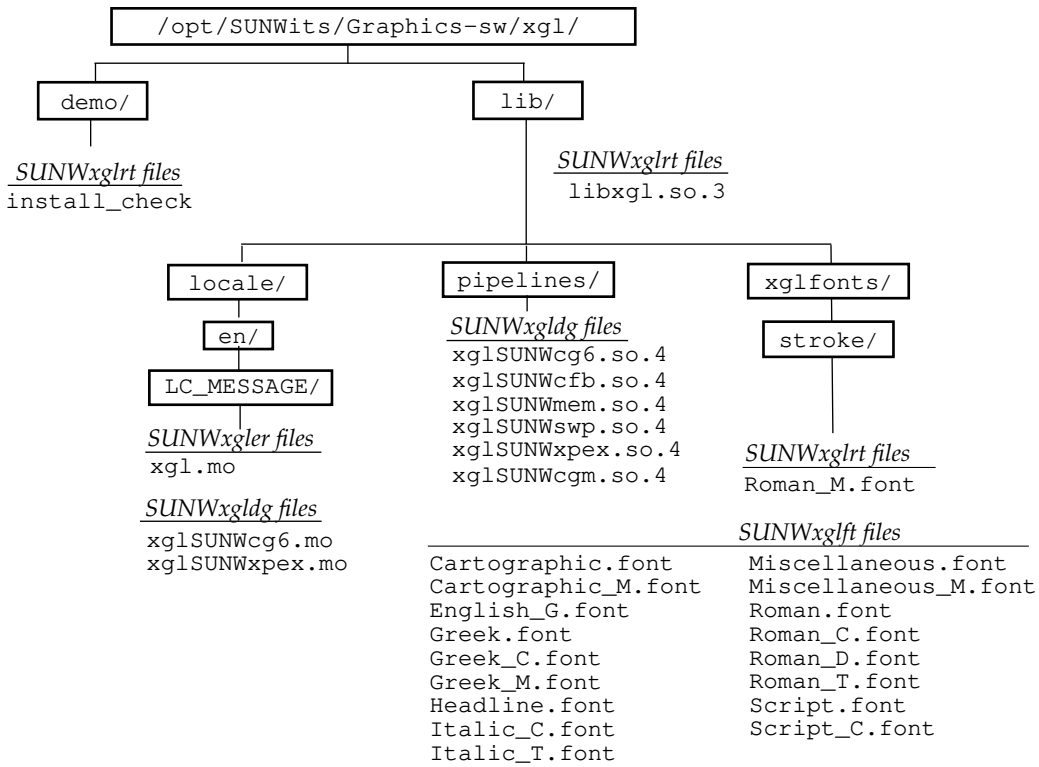

Figure 2-1 XGL Runtime Directory Structure

Font Files

The files in the `lib/xglfonts/stroke` directory contain the current XGL font files as well as links from the XGL 2.0 font file names, which are provided to accommodate programs written for the XGL 2.0 library. Links are from the XGL 2.0 font file names with the suffix `.phont` to the current XGL font files with the suffix `.font`.

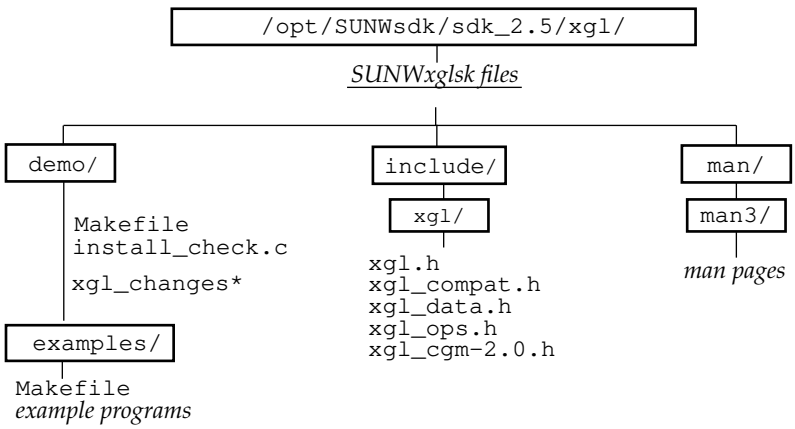

Figure 2-2 XGL SDK Directory Structure

XGL AnswerBook Directory Structure

Figure 2-3 shows the directory structure of the XGL AnswerBook package. The `_data` directory contains the files the AnswerBook software uses to display the XGL SDK documentation.

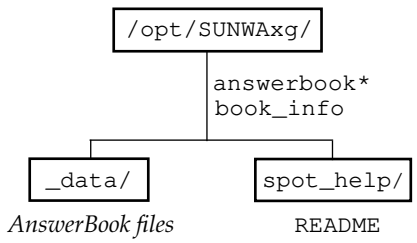

Figure 2-3 XGL Answerbook Directory Structure

Setting XGL Environment Variables

You need to set the XGLHOME and LD_LIBRARY_PATH environment variables to use the XGL product. These instructions assume that the XGL library is installed in the default directory.

1. **Set the XGLHOME environment variable to the directory containing the XGL library.**

```
host% setenv XGLHOME /opt/SUNWits/Graphics-sw/xgl
```

2. **Set LD_LIBRARY_PATH to include $XGLHOME/lib.**

```
host% setenv LD_LIBRARY_PATH $XGLHOME/lib:$LD_LIBRARY_PATH
```

Verifying the Installation of XGL

You can use the install_check program in the XGL demo directory to verify that the XGL product has been installed correctly. The install_check program allows you to list accelerated features for devices that support 8-bit and 24-bit rendering. install_check command line options are as follows:

```
install_check -color [RGB]  [index] -depth [24]  [8]
```

If you do not use the command line options, the program opens a 24-bit window with RGB color if a 24-bit visual is available; otherwise, the program opens a window with 8-bit indexed color.

To use install_check, follow these instructions:

1. **Change directory to the directory containing the XGL library.**
   If the XGL library was installed in the default directory, the command is:

```
host% cd /opt/SUNWits/Graphics-sw/xgl
```

   install_check expects to find the XGL library in the default installation directory and the OpenWindows library in /usr/openwin. If install_check cannot locate the XGL library, it uses the path listed in $LD_LIBRARY_PATH.

2. **Change directory to the** `demo` **directory and run** `install_check`**.**

```
host% cd demo
host% install_check
```

The `install_check` program displays a window labeled "XGL Installation Checker". The window contains XGL stroke text describing the device that XGL is using on your system. If the program runs successfully, your XGL library has been correctly installed.

Note – The `install_check` program uses a 24-bit visual if one is available; otherwise, it uses an 8-bit visual. If you try to mix 8-bit Xlib rendering on top of 24-bit XGL rendering, some devices will not display the Xlib rendering. In this case, start `install_check` with a depth of 8 to render over XGL graphics with 8-bit Xlib graphics, or render XGL graphics in an overlay plane.

Installation and System Administrator Notes

This section presents information on installation and system administrator issues.

XGL and Solaris

The XGL library runs within the current version of the Solaris product. This means that XGL is compatible with the component parts of the Solaris product. XGL is not supported with earlier versions of Solaris or its component parts.

Debuggable XGL Runtime Library

Due to space constraints, the XGL runtime environment installed with the operating system has been stripped of the symbol table information. This does not interfere with the normal running of an application built with the XGL library, but it may make it difficult for you to debug your graphics application, since you will not be able to produce stack traces from the `.so` files. Therefore, patch packages that contain XGL runtimes with all the symbol table information have been provided on the CD. See your system administrator for assistance if you need to have the patches installed.

Rendering Remotely Through PEX

If a window raster is created on a remote server that supports PEX, but the PEXlib library, `libPEX5.so`, cannot be found on the local machine, XGL issues an error message and uses Xlib to communicate with the remote window raster. To eliminate the error, either install the PEXlib library or disable PEX on the remote server.

If You Experience Poor Performance for Local Rendering

If you notice poor performance when rendering locally on an accelerated graphics device, the cause may be a security feature in Solaris that governs frame buffer access by processes that were started by users other than the first user of the window system. As a result of this security feature, non-owners of the window system do not have access to DGA. For example, if a user starts the window system, and then another user sits at the workstation, changes to his own environment using `su`, and starts an XGL application, the application will not run via DGA even though the second user is running the application locally. In this case, XGL uses Xlib to render even when XGL is running locally.

To give all local users access to DGA, follow these steps:

1. **Change directory to the file `/etc/logindevperm`.**
2. **Change the permissions to `0666` in the following lines.**

```
/dev/console   0666   /dev/mouse;/dev/kbd
/dev/console   0666   /dev/sound/*       # audio devices
/dev/console   0666   /dev/fbs/*         # frame buffers
```

Use this command:

```
host% chmod 666 /dev/mouse /dev/kbd /dev/sound/* /dev/fbs/*
```

3. **Reboot.**

2

Accessing the XGL Man Pages

To access the XGL reference manual pages, the `MANPATH` environment variable must be modified to include the directory where the man pages are located. `catman`(8) formats all the reference manual pages and stores the formatted version of each page. This optional operation causes the `man` command to work faster. It also builds the `whatis` database so that the keyword search option of `man` (`-k`) will include the XGL reference manual pages.

1. **Set the `MANPATH` environment variable to point to the directory where the XGL reference pages are located.**

```
host% setenv MANPATH /usr/man:/opt/SUNWsdk/sdk_2.5/xgl/man
```

2. **Format the man pages.**
   This will enable you to use the keyword search option (`-k`).

```
host% su root
Password: enter your root password
root% /usr/etc/catman -M /opt/SUNWsdk/sdk_2.5/xgl/man 3
```