X11, PostScript and Sun Microsystems' OpenWindows 3

This chapter discusses the OPEN LOOK development environment, the X Window System, Version 11 (X11) and the parts of OpenWindows 3.5 that implement aspects of X11.

OPEN LOOK

The OPEN LOOK Graphical User Interface (GUI) was developed by Sun Microsystems in partnership with AT&T. In July 1988, Sun and AT&T distributed more than 1000 copies of the OPEN LOOK specification draft to UNIX system users for review. The comments received from the industry were used to create the final version of the OPEN LOOK specification.

OPEN LOOK is a specification for a user interface within a window environment based on the pioneering work done on graphical user interfaces at Xerox PARC in the 1970s. A graphical user interface standard describes how applications appear on the screen and their behavior in relation to the user. The OPEN LOOK GUI was designed to provide a simple, consistent, and efficient interface.

The OPEN LOOK GUI uses windows and menus with common graphic symbols instead of typed system commands to provide an intuitive environment with a consistent screen layout that can be used across various platforms and operating systems.

3

OPEN LOOK compliance has two components: toolkit compliance and environment compliance. There are three types of software that fit within these two categories: toolkits, applications, and environments. Because most applications are built using toolkits, they usually assume the level of compliance characteristic of the toolkit used to create them. Toolkits, applications and environments as understood in OPEN LOOK parlance are described below:

- Toolkits. A toolkit is a set of programming components used to build OPEN LOOK GUI applications. It consists of a high-level programming interface that provides the elements required to build a user interface. Toolkits provide a set of routines that implement the various interface elements as defined by the specification. The application developer uses the routines provided by the toolkit to create and position the interface elements as needed. The toolkit makes application development easier and ensures that the user interface remains consistent by using the same building blocks. Toolkits help developers create user-interface prototypes by providing a simple and easy-to-use programming interface.
- Applications. An application is a program or set of programs designed to perform a specific task. Applications are built using a user-interface toolkit or other developer tools. While it is possible for the application developer to implement the OPEN LOOK User Interface (UI) without a toolkit, the usual approach is to use a toolkit written for a specific windowing platform. Together, applications and their use of the toolkit define the way programs look and feel to users.
- Environments. An environment is a program or set of programs that effect the design and operation of an OPEN LOOK GUI implementation. An OPEN LOOK compliant environment consists of an OPEN LOOK User Interface (UI) window manager, file manager, workspace properties window, and other utility programs.

To guarantee an OPEN LOOK GUI-compliant application, the developer must write the application with an OPEN LOOK GUI compliant toolkit and run the application in a compliant OPEN LOOK GUI environment.

Trademark licensing is available for the following three levels of OPEN LOOK certification:

- Level 1: This level contains all the essential components of a complete user interface. It delineates the minimum features required to certify an implementation as OPEN LOOK GUI compliant. Among the features covered in Level 1 are window types and properties, menu formats, and mouse and keyboard.
- Level 2: Compliance with Level 2 requires the presence of all of the features comprising Level 1 and additional features mandatory for Level 2. These include abbreviated buttons, nonstandard window types, scrollbars, and icon settings for color implementations.
- Level 3: Level 3 is a superset of Level 2. This level requires that an application contain certain specialized features and a process manager for extending the functionality of the OPEN LOOK GUI.

For a complete list of the required features for each level, see "Appendix A, Certification" in the *OPEN LOOK Graphical User Interface Functional Specification*.

Solaris Compliance With OPEN LOOK

The Sun GUI is a superset of OPEN LOOK and includes Sun value-added features. In moving toward full support of the Sun GUI, all required OPEN LOOK features will be supported by Sun at all levels.

OpenWindows implements the OPEN LOOK GUI standard. OpenWindows implements both the 2-D and 3-D OPEN LOOK GUI standard.

OpenWindows is a component of Solaris 2.5. The OPEN LOOK components indicated below implement the OPEN LOOK GUI.

- OPEN LOOK Intrinsics Toolkit (OLIT): The OPEN LOOK Intrinsics Toolkit was created by building a set of widgets for the X Toolkit Intrinsics (Xt) that conform to the OPEN LOOK GUI specification. The OPEN LOOK Intrinsics Toolkit API matches the X11 Release 4 Xt Intrinsics programming interface. For Solaris 2.5, OLIT is Level 2 compliant with certain exceptions.
- XView Toolkit (X11-based Visual/Integrated Environment for Workstations): XView is an X11 toolkit for building applications. The XView API is based upon Xlib, the lowest level of programming available to the X window system programmer. XView implements the OPEN LOOK GUI. For Solaris 2.5, XView is Level 2 compliant with certain exceptions.

- OpenWindows DeskSet: OpenWindows includes a set of OPEN LOOK applications that are collectively known as the DeskSet environment. All of the DeskSet applications support the OPEN LOOK model of dragging and dropping objects. The File Manager DeskSet tool is required by the OPEN LOOK standard for Level 2 compliance; it represents files (including directories and applications) with glyphs. The OpenWindows DeskSet File Manager program fulfills the Level 2 compliance requirement.

OPEN LOOK Specification and Related References

The specifications listed below address OPEN LOOK. They are published by Addison-Wesley.

- *OPEN LOOK Graphical User Interface Functional Specification* (ISBN 0-201-52365-5)
- *OPEN LOOK Graphical User Interface Application Style Guidelines* (ISBN 0-201-52364-7)

Licensing the OPEN LOOK Trademark

OPEN LOOK is a trademark of UNIX System Laboratories (USL), a wholly owned subsidiary of Novell, Inc. The Trademark License Agreement is the contract entered into by OPEN LOOK trademark applicants and USL. As a developer of OPEN LOOK applications, you may wish to license the OPEN LOOK trademark. USL has developed a trademark agreement as a formality to protect the trademark. To obtain the use of the OPEN LOOK trademark, follow the recommendations described below; no payment is required.

- For information on how to develop an OPEN LOOK application, refer to the *OPEN LOOK Graphical User Interface Functional Specification* that is delivered with OpenWindows.
- To receive your "OPEN LOOK Graphical User Interface Trademark License Agreement" forms, call 1 (800) 828-UNIX. If you are a UNIX system licensee, ask for your account representative. Otherwise, a sales associate will handle your request. After an authorized representative of your company signs the agreement, send it to USL at the following address:

UNIX System Laboratories
Attention: Sales Associate (or the name of your account representative)
PO Box 25000
Greensboro, NC 27420-5000.

Within 30 days, USL will send you an executed agreement authorizing you to use the OPEN LOOK trademark on your software.

X Window System, Version 11 (X11)

The X Window System, Version 11 (X11), developed by the Massachusetts Institute of Technology (MIT) X Consortium includes the following specifications: the Xlib C Language Interface (Xlib), the X Toolkit Intrinsics C Language Interface (Xt), and the Bitmap Distribution Format 2.1 (BDF).

X11 is a network-based protocol. A client application can run on the same or different system from the server that controls the display. In this server-client model, the application (which may run on one machine) is referred to as the window client. The system on which the user-interface is displayed may be a different machine and is referred to as the display host or window server.

Window systems are usually based on a pixel imaging model or a stencil/paint imaging model. The imaging model layer of the windows architecture controls how the window system accesses the display. X11 uses a pixel-based (raster) model in which images are viewed as rectangular areas of device-dependent pixels.

The Xlib library routines communicate with the X11 server via the X protocol. Xlib is the lowest-level C language application programming interface (API) to the X protocol.

The main task of Xlib is to translate C data structures and procedures into X protocol events; it sends them off and receives protocol packets in return that are unpacked into C data structures. Xlib provides full access to the capabilities of the X protocol but does little to make programming easier. It handles the interface between an application and the network and includes some optimizations that encourage efficient network usage.

Because application development at the Xlib level can be tedious, MIT developed the X toolkit, Xt. The designers of Xt were aware that the toolkit would need to support a variety of graphical user interface standards. For this

reason, Xt was divided into two portions. The first portion is a prebuilt set of user interface components known as widgets. The second portion is the programmer interface for manipulating widgets, known as intrinsics.

Although it is device-independent, X11 allows an application to tailor itself to the hardware on which it is run.

PostScript Language

The PostScript™ language, from Adobe Systems Inc., is the modern standard for electronic printing. The first edition of the *PostScript Language Reference Manual*, published in 1985 by Addison and Wesley, established PostScript Level One. Today, PostScript is supported as a standard by all major computer, printer, and imagesetter vendors.

Numerous extensions were requested by the industry, so, in 1990, the second edition of the reference manual was published. It describes three major extensions to PostScript Level One: (1) An extension to deal with color output, (2) A composite font model, mainly used for very large fonts (for example, Asian languages) and (3) A set of extensions for screen output, called the Display PostScript™ system, or DPS. DPS displays graphical information on the computer screen with the same imaging model and PostScript language that are the standards for printers and typesetters.

These major extensions and a large number of minor ones comprise PostScript Level Two, often referred to as PS2 or PSL2.

Solaris Compliance With X11

OpenWindows consists of the OpenWindows server, the Display PostScript™ (DPS) extension support, the OpenFonts™ Technology, the OPEN LOOK window manager (olwm), the XView Toolkit, the OPEN LOOK Intrinsics Toolkit (OLIT), the DeskSet™ tools, and demonstration applications.

The OpenWindows server and the associated X libraries, which include Xlib and Xt, are compliant with X11, Release 5.

The OPEN LOOK Intrinsics Toolkit API is an implementation of MIT's Xt toolkit with an OPEN LOOK widget set. AT&T created the OPEN LOOK Intrinsics Toolkit by building a set of widgets for Xt that conform to the OPEN LOOK GUI specification.

The XView toolkit (X Window System-based Visual/Integrated Environment for Workstations) is a C-language toolkit providing a rich set of components for building applications. Like the Intrinsics, XView is built on Xlib. SunSoft has made the source code to the XView Toolkit freely available. It is shipped as part of the standard MIT X distribution and with UNIX System V Release 4.

OpenWindows, through the OPEN LOOK window manager, fully supports the X11 Inter-Client Communications Conventions (ICCC) as defined in X11 Release 5. The ICCC manual provides basic policy intentionally omitted from X itself, such as rules for transferring data between applications, transfer of keyboard focus, layout schemes, colormap installation and other features.

Solaris Compliance with PostScript

The OpenWindows server is a complete implementation of PostScript Level Two. The Display PostScript system is implemented as an extension to the X Window System and includes the following enhancements:

- Support for F3 Latin and Asian fonts
- Support for obtaining prescaled bitmap font formats from X11 font code

X11 Specification and Related Publications

The first publication listed below defines the X11 protocol specification; it is also defined in subsequent supplements supplied with X11 Release 5.

- *X Window System Third Edition*, Schiefler & Gettys, Digital Press, 1992
- *XView Programming Manual*, O'Reilly & Associates, Inc., 1989
- *XView Reference Manual*, O'Reilly & Associates, Inc.
- *PostScript Language Reference Manual*, *Second Edition*, Addison-Wesley
- *XView Developer's Notes*, Sun Microsystems, Inc.
- *OpenWindows Desktop Reference Manual*, Sun Microsystems, Inc.
- *Desktop Integration Guide*, Sun Microsystems, Inc.

3