Building X Into the Network

X Is Maturing Fast — Flourishing in the Workstation/Server Environment

BY BRADFORD T. HARRISON

The X Window System defines the basic mechanism to implement Graphical User Interfaces (GUIs) in the distributed computing environment. X was designed as part of Project Athena at MIT, and due to strong support from the Open Software Foundation (OSF) and its members, as well as Sun Microsystems and a groundswell of X-specific third-party companies, X has rapidly become the de facto standard for implementing GUIs.

GUIs are becoming as fundamental to computing as computer architectures, operating systems, and networking protocols. In the same way that sophisticated multitasking operating systems replaced primitive collections of low-level device drivers, X is replacing the command line interface.

The UNIX environment is, for the most part, the platform used by X application and system software developers, and a wide range of X-based applications and system software are becoming available. The software products "fit" at different layers of the X model (see *Figure 1*), and a user selects products based on factors such as the style of interface he requires and at what layer of the X model he needs to implement X.

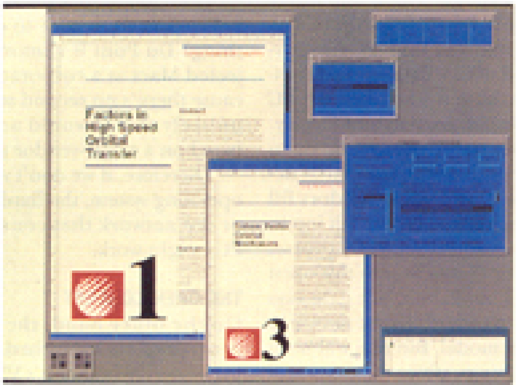
WORKSTATIONS UNBOUND

Competitive products addressing all layers of the X model are rapidly becoming available. A core group already exists. Products from the same vendor tend to be available on all major UNIX platforms, including platforms from HP Apollo, Digital, Sun, and IBM. Though they are usually developed on the platform that the company's software engineers are most familiar with, X software products are easily ported across UNIX-based systems.

But on the client (application) side, X requires huge amounts of memory and processor power. In addition, X servers (displays located anywhere on the network) need plenty of memory to avoid excessive interaction across the network with a client due to limited local resources. The large memories and strong mips ratings of today's workstations are proving necessary to support the use of X. The PC, at least in its traditional form, will have a hard time keeping up, but it's certainly trying. (See Dressing Up The PC And MAC, page 15)

X is growing fast — much faster than anyone anticipated. The limitations of X, in fact, are already becoming apparent. The main limitation is that X, as a distributed computing mechanism, doesn't do much more than provide the graphical interface between the application and the user or users using the application. It is only concerned with the semantics of the interface.

Furthermore, since most users run X from a workstation with its own powerful processor, it's a waste to use such a powerful machine only for viewing. The full resources of the workstation need to be exploited in the distributed X environment. This will be accomplished by integrating X with the dominant RPC mechanisms available from Sun/AT&T and OSF, as shown in *Figure 2*. Companies such as Visual Edge (Montreal, Canada) have already begun work in this area.



Solbourne's C++-based OI library. A single application is invoked twice on the same screen using both the Open Look and Motif GUIs

MANAGING THE X DESKTOP

Unless you're happy using a UNIX shell from the command line inside a window, you'll want to implement an X desktop manager product for your specific environment. The best known X-based desktop products are available from Visix Software (Reston, Va.), IXI Ltd. (Cambridge, England), and Hewlett-Packard.

Desktop managers isolate the user from the specifics of the underlying operating system. Applications and files appear as icons, and commands are accessed via pull-down menus. A desktop manager is built into the Macintosh operating system, whereas it must be added to UNIX machines.

Desktop manager products can be distinguished on the basis of the nature of the interface to the underlying operating system (they can be completely disguised or only partially hidden) as well as support for existing applications and file types. For example, the Looking Glass product from Visix attempts to retain the "feel" of the UNIX shell and supports some 450 file types. Furthermore, since Visix has written its own X toolkit, Motif, Open Look, Presentation Manager, and other GUIs are supported. The GUI is selected by the user at runtime.

On a Sun machine, Looking Glass incorporates older SunView applications into the X environment. SunView was the first toolkit developed at Sun and was not X-based. Now most users are migrating away from SunView to Sun's XView toolkit, which is based on X. Looking Glass was originally developed in the SunView environment and now (in addition to supporting all the major UNIX platforms) is easing migration pains from SunView to XView for users of older Sun workstations.

OPEN LOOK VS. MOTIF

The hottest debate in the X world today is between the two most popular X GUIs, Open Look and Motif. Open Look is supported by Sun and AT&T, while OSF and its member companies are solidly behind Motif. The third parties usually support both.

Support for both can occur in two ways: One, an X application uses calls to either a Motif or an Open Look toolkit, and a version of an application of each type is offered by the company; Two, a special toolkit is used that supports "generic" calls independent of the specific interface, allowing the user to select the interface he wants at runtime.

Solbourne's (Longmont, Colo.) Object Interface (OI) library is the best known toolkit supporting the latter method. The photo shows how using this toolkit enables a user to run applications using both Open Look and Motif at the same time. However, mixing the "look and feel" of both on the same display is odd and is not recommended by Solbourne.

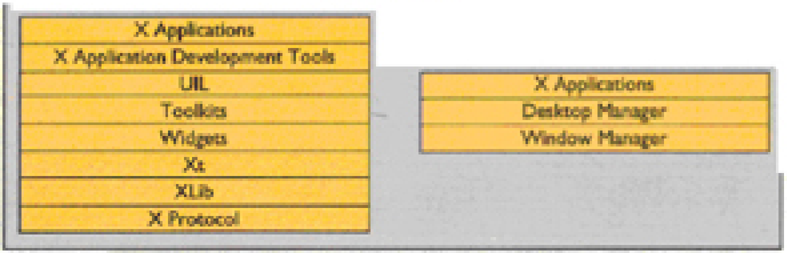
Toolkits supporting both are easily ported across UNIX platforms. TGV (Santa Cruz, Calif.) has even ported the Sun XView toolkit to a non-UNIX operating system, Digital's VMS, allowing Open Look applications originally written for Sun machines to run on the VAX. The main Motif toolkit is OSF's, while Open Look is usually implemented with either XView from Sun or the Open Look Intrinsics Toolkit (OLIT) from AT&T.

BUILDING GUIs

User Interface Management Systems (UIMS) are used to develop user interfaces. In the X world, two UIMSes have charged to the forefront — the UIM/X product from Visual Edge, and the Open Windows Developer's Guide from Sun. UIM/X supports both Open Look and Motif, while Sun's product is Open Look specific. Other less-generic UIMS products used to develop X applications are available from ICS (Cambridge, Mass.), TeleSoft (San Diego, Calif.), and Non-Standard Logics (Paris, France).

Using a GUI development tool allows programmers to quickly prototype and test an interface to their application. According to Hewlett-Packard, generating the interface to an X application can require as much as 50 percent to 80 percent of a programmer's time, but with these tools he can quickly build the interface interactively. The source code is automatically generated for him. Interface design can even be turned over to people with backgrounds in ergonomics, cognitive psychology, and human-machine interaction — people more qualified to construct an appropriate user interface than the programmer himself.

FIGURE 1

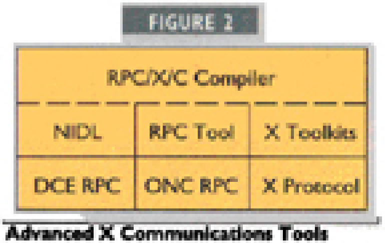


X Layering

Besides interface building, GUI builders are providing a variety of other capabilities. For example, the UIM/X product allows developers to build libraries of templates as various "looks" are defined for later use. In this way, developers can reuse the look they like, or share the look with other programmers, providing a consistent user interface across a base of applications.

UIM/X also allows developers to "retrofit" traditional applications into the X environment. The product provides the hooks to achieve this to different degrees, ranging from equipping an application with a simple windowing interface to establishing a primitive event loop within the application. The second approach is more difficult and involves recoding, while the first can be accomplished with no changes to existing code.

FIGURE 2



Advanced X Communications Tools

UIM/X also provides an interface to User Interface Language (UIL), a language developed by Digital to ease application development for DECwindows X-based applications, and which was adopted by OSF as part of Motif. UIL, like UIM/X, makes tasks easier to perform than if they were coded using the X toolkit directly.

COOPERATION AND INTEGRATION

Visual Edge doesn't sell directly to customers, but it has OEMed UIM/X to many companies including Digital, IBM, HP, Motorola, Silicon Graphics and AT&T. This activity epitomizes the phenomenon of X: Third parties are no longer "third party" to a specific large company — they are becoming third parties to all the giants. With the proliferation of open systems, third parties don't exist in the shadows of the giants, they directly participate and enhance the products of the larger companies and thrive in their own areas of expertise.

For example, HP's implementation of UIM/X, the HP Interface Architect, uses the core functionality of UIM/X, but HP worked with Visual Edge to build its own custom features into the product. The product is Motif-specific and is integrated with the other components of NewWave, HP's strategy to tie together the disparate elements of its UNIX, Apollo, and proprietary systems environments.

Interface Architect runs in the SoftBench environment, HP's collection of CASE products, and the applications produced are used with HP's Visual User Environment (VUE) desktop manager. All of HP's workstation products, however, are based on UNIX, and the hooks between the products don't require customers to implement the full suite of HP products if they like just one. For example, applications developed with Interface Architect will run on any platform supporting the OSF/Motif toolkit, and with any desktop manager. HP's strategy is to support standards while attempting to provide the best specific product implementations within each standard. Its Motif-based products are a good example.

With the portability of X and the rapid growth in application development tools, the dearth of X applications available for a variety of machines is rapidly becoming history.

Arguments against X, such as its resource-hogging and the difficulty in restructuring existing applications to operate under the "event loop" required by X-based applications, continue to plague widespread application of the standard.

Cheap, character-oriented terminals attached to a minicomputer remain the least expensive solution for many applications. But all revolutions take time, and there can be no doubt that the X revolution is solidly underway.

DRESSING UP THE PC AND MAC

In the rush to GUIs, the PC and Macintosh are certainly not going unadorned.

Windowing started with the Mac, and now it's participating in the X environment as well. Using products from Apple, MacTCP and MacX, the Macintosh can serve double-duty as a desktop computing device and a full-fledged X server in UNIX-based workstation/server networks. White Pine Software (Amherst, N.H.) also sells a product for the Mac, eXodus, that turns it into an X server.

The dominant GUI for the PC is Microsoft Windows. Windows for the PC is meant more to compete with the Macintosh than UNIX workstations. Windows was designed to take advantage of the 80286 and 80386 microprocessors from Intel, and provides a Mac-like desktop manager. Windows also provides the ability to connect to file servers from within the Windows environment, and to print remotely.

In addition, such products from Locus (Inglewood, Calif.), Graphic Software Systems (Beaverton, Ore.), and Quarterdeck Office Systems (Santa Monica, Calif.) turn a PC into an X server. When used with one of the several TCP/IP implementations for the PC, these products turn PCs into useful, though slow, X servers on networks of UNIX workstations. In addition, DESQview X, to be introduced later this year from Quarterdeck, will turn a PC into a multitasking DOS system with full X server and client capabilities.

But UNIX is also available on the PC and Mac. The Santa Cruz Operation's UNIX (SCO UNIX) is supported by several X vendors, including Visix, which offers the Looking Glass desktop manager for SCO PCs. Apple's AUX UNIX operating system for the Mac is also finding support, though Apple itself is dragging its feet in the UNIX marketplace.