Solaris VISUAL

A White Paper

SunSoft
A Sun Microsystems, Inc. Business

Contents

Introduction 1

The graphics industry has historically driven computing innovation because of the demands it places on computing hardware and software. Simultaneously, graphics standards have emerged which simplify the development effort required to build volume applications. The Common Open Software Environment is an example of the computer industry focusing on standards—with the goal of interoperability of open systems. Among other things, this environment defines as set of important standard interfaces for graphics rendering.

Despite the emergence of industry and de facto standards for graphics application program interface (API) libraries, new technology and new markets will likely continue to spawn new interfaces. In effect, there will always be innovation. And with advances in multimedia, object-oriented programming, integrated graphics and emerging applications such as virtual reality, it is clear that this innovation will continue to remain important.

Together with standards and innovation, users have come to expect performance and functionality in graphics, which address their widely varied requirements. SunSoft has developed Solaris© VISUAL™ with these considerations in mind: standards, innovation, performance and functionality.

SunSoft sees the evolving graphics marketplace as one becoming increasingly meshed with general purpose computing, where user's requirements for graphics will become broader and more widespread. As higher-end applications such as document imaging, collaborative multimedia and 3-D graphics become mainstream, they will move to the volume desktop and enjoy much wider use.

To address the trend in desktop graphics, SunSoft has developed a strategy which capitalizes on industry trends in standards and users' requirements for choice. Whether the user is selecting a graphics library for application development or just an application, Solaris VISUAL enables and encourages a wide selection of graphics solutions. Solaris VISUAL provides a flexible graphics environment upon which standards can interoperate and innovation can prosper.

The Demands of Graphics

As hardware platforms become faster, applications also become more demanding. New features, broader functionality, increased quality and distributed computing facilities all contribute to the processing demands of graphics applications.

Increased rendering quality is a major trend in graphics. New applications which provide imaging, 3-D geometry, animation and more realistic display are becoming common.

Another significant development in computing is the move toward distributed client-server computing. The flexibility of client-server computing in providing users with resources on demand, adjusting to organizational restructuring and offering lower cost has made it the preferred approach for the majority of businesses today.

SunSoft's vision of distributed computing with graphics includes interconnected networks of many computing technologies. The spectrum of compute resources ranging from mainframes and minicomputers to servers, workstations and PCs will enjoy greater connectivity and integration. From a graphics application perspective, this means that users may manipulate data from virtually any source, and be able to display and interact with it from any desktop platform.

SunSoft's foundation approach to graphics lends itself well to the enterprise computing model, where X terminals, PCs and workstations will work in concert with mainframes and supercomputers. X Windows-based applications running on mainframes will display on desktop systems, regardless of their location on the network. Data servers can provide data to supercomputers for internal rendering, with display to PCs and X terminals across the building or across the world.

Users and Graphics

As users employ an increasingly large array of graphics applications, they are demanding richer graphics functionality for display of window systems, images, geometry and digital video. Each of these graphics areas has differing performance and display requirements, and for the developer of applications, there are different application program interface needs.

Equally important is the need to provide seamless integration of graphics applications, so that imaging applications can display, even share data, with geometry applications.

Users today need to be able to manipulate graphics information on a diverse range of platforms, and be able to perform universal rendering to any device in a distributed fashion. Only a uniform model for graphics rendering will make this possible.

Above all, users need to realize the benefits of graphics applications. Information content can be increased provided applications run well and quickly, with good response time for user input and graphics display.

Solaris VISUAL is a uniform, integrated, scalable graphics foundation, providing the flexibility, features and performance of a broad range of graphics applications. Developers have fundamental freedom of choice in selecting the right interface to meet their needs. And users can be confident that their applications will run on any Solaris-based platform.

Application Software Developers

Developers of graphics software require rich functionality in order to target specific market areas with their application. For example, the needs of a developer of electronic design applications are different than those of a color prepress application developer. These same developers need their investment in application porting to be as well preserved as possible. Hence, developers need graphics interfaces that can grow and evolve, without incompatible changes to the API.

Software developers also need device independence built into the graphics API. The effort necessary to support differing graphics devices from the same application places unnecessary burden on the developer and increases the cost of development, maintenance and support.

Finally, developers need interfaces that provide the features and functionality they require for their application. High performance, rendering intensive applications need optimized high performance APIs which can take advantage of available graphics hardware devices as well as provide runtime optimization based upon available system characteristics.

Solaris VISUAL addresses the needs of developers through optimized foundation interfaces which provide the features and performance needed by today's applications. Most Solaris VISUAL interfaces are object-oriented to hide the underlying details of their implementations, enabling them to gracefully evolve, while remaining compatible with previous releases.

Solaris VISUAL

Solaris VISUAL provides the base-level functionality and flexibility to support integrated high-performance graphics, imaging and digital video applications, as well as the Application Programming Interfaces (APIs) they may be implemented on. Solaris VISUAL consists of three key parts: Foundation Libraries, X Graphics Libraries and Graphics Porting Interfaces (GPIs).

Solaris VISUAL is the first UNIX-based computing environment to open its graphics environment completely, thus providing ease of integration, simplified installation, and complete compatibility between hardware display devices and applications.

Solaris VISUAL is the most complete set of foundation graphics display tools available today, and together with Solaris OpenWindows addresses application requirements for:

- *Geometry*
- *Imaging*
- *PostScript Rendering*
- *Pixel-based Rendering*

Figure 1.1 illustrates the principal graphics areas addressed by the Solaris VISUAL graphics environment components.

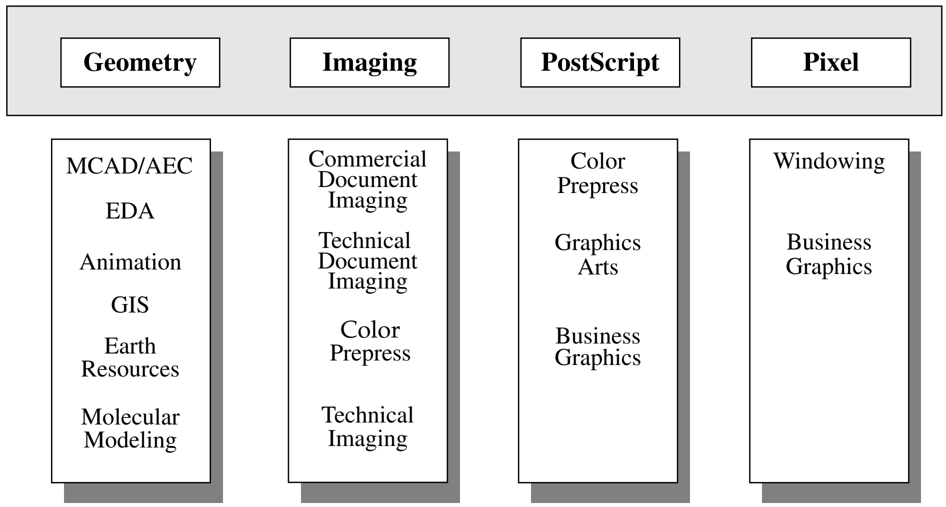

Figure 1.1 Solaris VISUAL Foundation Addresses a Broad Range of Graphics Markets

Solaris VISUAL bring open systems to the world of graphics, while addressing the needs of the volume software and hardware markets. Users win with Solaris VISUAL because their choices are greater, and their potential solutions are more numerous.

Solaris VISUAL Components

Figure 1.2 illustrates the Solaris VISUAL architecture. Solaris VISUAL has four foundation libraries corresponding to the above four major areas of graphics functionality.

XGL is a mature, feature-rich 2-D and 3-D geometry-oriented foundation interface. XIL contains a set of key functions from the fields of image processing and manipulation, which address applications and APIs for imaging and related markets. PostScript provides de facto standard PostScript Level 2 rendering, and X is the industry standard 2-D protocol common to all X Windows environments.

Above the foundation libraries, there are a set of standard X graphics libraries, one for each graphical area. These are PEXlib, XIElib[1], DPSlib, and Xlib. The foundation and X graphics libraries meet the key requirements of graphics-oriented applications and layered third-party APIs. Solaris VISUAL is a excellent platform for interfaces like PHIGS, GKS, HOOPS, OpenGL, PIKS and many others. Figure 1.2 illustrates the Solaris VISUAL components.

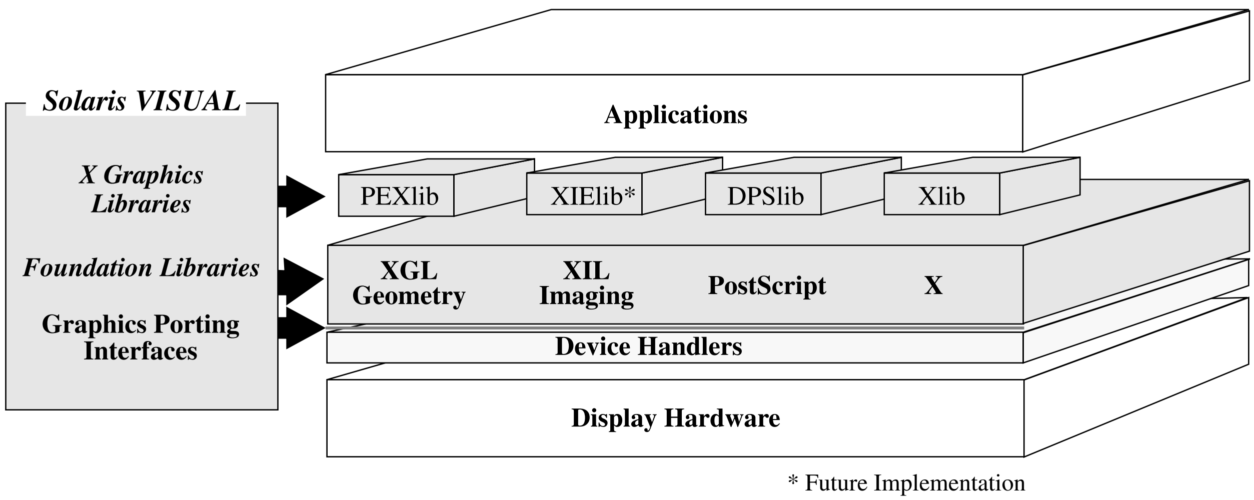

Figure 1.2 Solaris VISUAL Components

A key feature of SVR4 is its support for dynamic linking and loading, which enables low-level foundation interfaces such as XGL and XIL to dynamically load device handlers associated with underlying display hardware. In this way, devices such as framebuffers, video boards and geometry accelerators can easily be incorporated into Solaris-based x86 and SPARC platforms. The Solaris VISUAL Graphics Porting Interfaces (GPIs) simplify the development and delivery of device handlers which take advantage of dynamic linking. Stable GPIs permit IHVs to develop optimized device handlers for their hardware.

Dynamic linking also eliminates issues associated with applications being tied to specific operating system releases because applications will load the latest library.

1.XIElib is currently under development by the MIT X Consortium

Benefits of the Solaris VISUAL Graphics Environment

The architecture of the Solaris VISUAL Graphics Environment provides a layered interface scheme which addresses a variety of devices, with alternative resolution, rendering speed and functionality. Stable interfaces further enable hardware developers to enhance their products without concern that the interfaces will change. Finally, the device remains compatible across operating system releases and regardless of enhancements to the Solaris VISUAL foundation graphics interfaces, high level graphics APIs or the applications themselves.

This layered approach enables integrated graphics functionality because XIL, XGL, PostScript and X rendering layers access the window system in the same way. Applications which utilize XIL for imaging can generate geometry through XGL to the same window. And applications manipulating geometry can overlay raster images above the geometric data.

The standardization of the underlying device porting layers makes it easier for OEMs and IHVs to write graphics drivers, due to the relative ease in porting the foundation graphics interfaces to them. Additionally, software applications can become shrink-wrapped because they will work with any display device which supports the Solaris VISUAL graphics interfaces. An increased number of products, both hardware and software, will then drive a competitive market for Solaris-based graphics devices leading to lower cost products. Figure 1.3 illustrates how Solaris VISUAL addresses a wide range of platforms, devices, and applications.

The Graphics Hardware Developer (OEM, IHV)

The benefits to hardware developers reside in the ability to selectively invest in tuning their devices to individual Solaris VISUAL graphics interfaces. A hardware vendor must do a basic port to the X Foundation library to get the Solaris VISUAL environment running on a new device. The vendor may then choose to do optimized ports for one or more of the other foundation libraries. For example, a hardware developer may wish to optimize performance of both XIL and XGL rendering, and thus address multiple market segments (and users) with the same product.

Hardware developers have the added advantage of supporting any Solaris application, and do not need to concern themselves with developing drivers for individual applications. Any API built on top of Solaris VISUAL foundation graphics interfaces will run on the vendor's device, providing even broader market coverage.

The Graphics Library Developer

The graphics library developer (i.e. developer of graphics libraries such as HOOPs, etc.) need not target specific devices, develop drivers or otherwise be concerned about maintenance and support for individual graphics devices for their product. The Solaris VISUAL Graphics Environment specifies clear interface definitions at both the foundation interface and graphics porting interface (GPI) levels, thus insuring compatibility from the application all the way down to the device.

ISVs can offer their product on a Solaris computing platform and be secure in the knowledge that it will run on any Solaris-compatible graphics hardware.

The Application Developer

Solaris VISUAL gives application developers the freedom to select the right API for their needs. Instead of using only APIs provided by systems (hardware & software) vendors, application developers can select the API that matches their application's architecture, and meets their requirements for functionality, portability, flexibility and price.

The Enduser

For the enduser, the benefits of Solaris VISUAL truly come together. Endusers can select the application they need and run it on the graphics device they want, all according to their price and performance needs.

Further, the enduser can select a standardized computing platform, including the operating system, networking, development environment and administrative tools, and know that applications will always run. In selecting a standard platform, the task of systems administrators is simplified - one environment, one platform, and thousands of compatible applications.

Individual users can base their computing environment on a standard platform, but still retain the ability to tailor their environment according to their needs for applications, APIs and graphics devices.

This white paper describes the individual Solaris VISUAL components and their integration into the Solaris environment and how applications using Solaris VISUAL can use the distributed computing capabilities of Solaris.

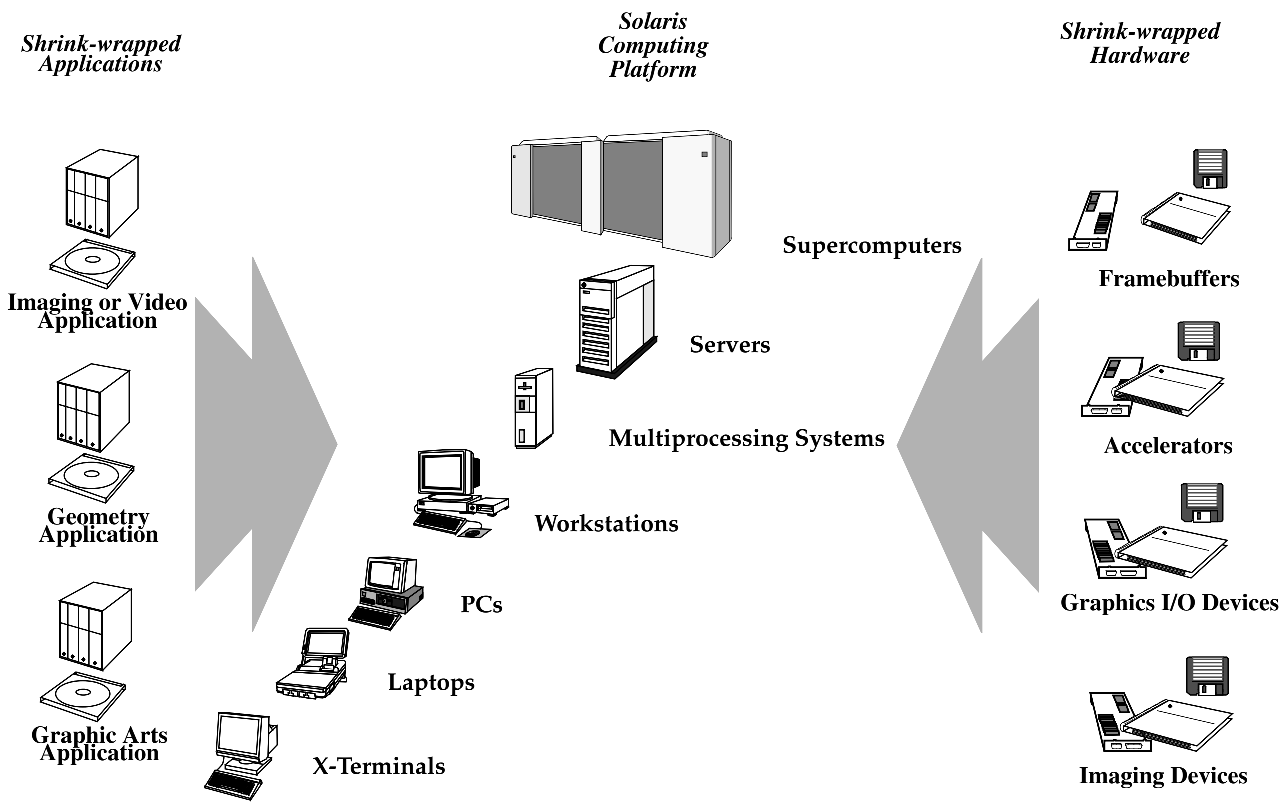

Figure 1.3 Solaris VISUAL Addresses a Broad Range of Platforms