

An API for Interfacing Interactive 3D Applications to High-Speed Graphics Hardware

Kevin T. Lefebvre

John M. Brown

The OpenGL[®] specification defines a software interface that can be implemented on a wide range of graphics devices ranging from simple frame buffers to fully hardware-accelerated geometry processors.



Kevin T. Lefebvre

A senior engineer in the graphics products laboratory at the HP Workstation Systems Division, Kevin Lefebvre is responsible for the OpenGL architecture and its implementation and delivery. He came to HP in 1986 from the Apollo Systems Division. He has a BS degree in mathematics (1976) from Carnegie-Mellon University. He was born in Pittsfield, Massachusetts, is married and has two children. His hobbies include running, biking, and skiing.



John M. Brown

John Brown is a senior engineer in the graphics products laboratory of the HP Workstation Systems Division. He is responsible for graphics application performance. John came to HP in 1988. He holds a BSEE degree (1980) from the University of Kentucky.

OpenGL is a specification for a software-to-hardware application programming interface, or API, that defines operations needed to produce interactive 3D applications. It is designed to be used on a wide range of graphics devices, including simple frame buffers and hardware-accelerated geometry processor systems. With design goals of efficiency and multiple platform support, certain functions, such as windowing and input support, have not been defined in OpenGL. These unsupported functions are included in support libraries outside the core OpenGL definition.

OpenGL is targeted for use on a range of new graphics devices for both UNIX[®]-based and Windows[®] NT-based operating system platforms. These systems differ in both capabilities and performance.

Early in the OpenGL program at HP, industry partnerships were established between the OpenGL R&D labs and key independent software vendors (ISVs) to ensure a high-quality, high-performance product that met the needs of these ISVs. These partnerships were also used to assist the ISVs in moving to the HP OpenGL product (see “The Fast Break Program” on page 8).

The various OpenGL articles in this issue describe the design philosophy and the implementation of the HP version of OpenGL and other graphics products associated with OpenGL.

History of OpenGL

OpenGL is a successor to Iris GL, a graphics library developed by Silicon Graphics® International (SGI). Major changes have been made to the Iris GL specification in defining OpenGL. These changes have been aimed at making OpenGL a cleaner, more extensible architecture.

With the goal of creating a single open graphics standard, the OpenGL Architecture Review Board (ARB) was formed to define the specification and promote OpenGL in terms of ISV use and availability of vendor implementations. The original ARB members were SGI, Intel, Microsoft®, Digital Equipment Corporation, and IBM. Evans & Sutherland, Intergraph, Sun, and HP were added more recently. For more information on current ARB members, OpenGL licensees, frequently-asked questions, and other ARB related information, visit the OpenGL web site at <http://www.opengl.org>.

The initial effort of the ARB was the 1.0 specification of OpenGL, which became available in 1992. Along with this specification was a series of conformance tests that licensees needed to pass before an implementation could be called OpenGL. Since then the ARB has added new features and released a 1.1 specification in 1995 (the HP implementation is based on 1.1). Work is currently being done to define a 1.2 revision of the specification.

HP Involvement in OpenGL

HP became an OpenGL licensee in 1995. We had the goal of delivering a native implementation of OpenGL that would run on hardware and software that would provide OpenGL performance leadership.

Shortly after licensing OpenGL, we established a relationship with a third party to provide an OpenGL implementation on our existing set of graphics hardware while we worked on a new generation of hardware that was better suited for OpenGL semantics. The OpenGL provided by the third party used the underlying graphics hardware acceleration where possible. However, it could not be considered an accelerated implementation of OpenGL because of features lacking in the hardware.

In August of 1996, we demonstrated our first native implementation of OpenGL at Siggraph 96. This implementation was fully functional and represented the software that

would be shipped with the future OpenGL-based hardware. The implementation supported various device drivers including a software-based renderer. The OpenGL development effort culminated in the announcement and delivery of OpenGL-based systems in the fall of 1997.

Software Implementation

In our implementation, we focused on the hardware's ability to accelerate major portions of the rendering pipeline. For the software, we focused on its ability to ensure that the hardware could run at full performance. A fast graphics accelerator is not needed if the driving software cannot keep the hardware busy. The resulting software architecture and implementation was designed from a system viewpoint. Decisions were based on system requirements to avoid overoptimizing each individual component and still not achieve the desired results. An overview of the HP OpenGL software architecture is provided in the article on page 9. Another software-related issue is provided in the article on page 35, which discusses issues associated with porting a UNIX-based OpenGL implementation to Windows NT.

Hardware Systems

The new graphics systems are able to support OpenGL, Starbase, PHIGS, and PEX rendering semantics in hardware. Being able to support the OpenGL API means that there is hardware support for accelerating the full feature set of OpenGL instead of just having a simple frame buffer in which all or most of the OpenGL features are implemented in software. These systems are the VISUALIZE fx2, VISUALIZE fx4, and VISUALIZE fx6 graphics accelerator products. These systems differ in the amount of graphics acceleration they provide, the number of image planes, and the optional OpenGL features they provide. In addition to the base graphics boards, a texture mapping option is available for the fx4 and fx6 accelerators. The article on page 28 provides an overview of the new graphics hardware developed to support OpenGL.

Engineering Process

To meet the required delivery dates of OpenGL with a high level of confidence and quality, we used a new process to compress the time between first silicon and manufacturing release. The article on page 41 describes the

The Fast-Break Program

In basketball, a rapid offensive transition is called a fast-break. The fast-break program is about the transition game for OpenGL on HP systems. A key part of the HP transition to OpenGL is applications, because applications enable volume shipments of systems. Having the right applications is necessary for a successful OpenGL product, but it is also important that the applications run with outstanding performance and reliability. Fast-break is about both aspects—getting the applications on HP systems and ensuring that they have outstanding performance and reliability.

Fast-break began by working with application developers in the early stages of the OpenGL program to understand their requirements for the HP OpenGL product. These requirements helped to drive the initial OpenGL product definition.

As the program progressed, the Fast-break team developed a suite of tools that enabled detailed analysis of OpenGL applications. Analysis of key applications was used to further refine our OpenGL product performance and functionality. Analysis also yielded a set of synthetic API benchmarks that represented the behavior of key applications. These synthetic benchmarks enabled HP to perform early hands-on evaluation of the OpenGL product long before the actual applications were ported to HP.

Pre-porting laid the groundwork for the actual porting of applications to HP's implementation of OpenGL. The first phase of

the porting took place during the OpenGL beta program. In this program, the HP fast-break team worked closely with selected application developers to initiate the porting effort. A software-only implementation of the OpenGL product was used, which enabled the beta program to take place even before hardware was available.

As hardware became available, the beta program was superseded by the early access program. This program included the original beta participants and additional selected developers. In both the beta and early access programs, HP found that the homework done earlier by the fast-break team paid big dividends. Most applications were ported to HP in just a few days and, in some cases, just a few hours!

Although not completely defect-free, these early versions of OpenGL were uniformly high-performance and high-quality products. By accelerating the application porting effort, HP was able to identify and resolve the few remaining issues before the product was officially released.

The ongoing involvement of the fast-break team with the OpenGL product development teams helped HP do it right the first time by delivering a high-quality, high-performance implementation of OpenGL and enabling rapid porting of key applications to the HP product.

engineering process we used to accelerate the time to market for OpenGL.

Graphics Middleware

A fast graphics API is not always enough. Leading edge CAD modelling problems far exceed the interactive capacity of graphical super workstations. For example, try spinning a complete CAD model of a Boeing 777 at 30 frames per second on any system.

What is needed is a new approach to solving the rendering problem of very large models. The goal is to trade off between frame rate, image quality, and system cost.

HP has introduced a toolkit for use by CAD ISVs to assist them in solving this problem. The toolkit is called DirectModel and is described on page 19.

HP-UX Release 10.20 and later and HP-UX 11.00 and later (in both 32- and 64-bit configurations) on all HP 9000 computers are Open Group UNIX 95 branded products.

UNIX is a registered trademark of The Open Group.

X/Open is a registered trademark and the X device is a trademark of X/Open Company Limited in the UK and other countries.

Microsoft is a U.S. registered trademark of Microsoft Corporation.

Windows is a U.S. registered trademark of Microsoft Corporation.

Silicon Graphics and OpenGL are registered trademarks of Silicon Graphics Inc. in the United States and other countries.

-
- ▶ [Go to Next Article](#)
 - ▶ [Go to Journal Home Page](#)