SimGL™ And OpenGL®

How SimGL Optimizes OpenGL for Interactive 3D Graphics Applications

Background and History

This white paper focuses on the Quantum3D SimGL API. SimGL, although identical in many ways to the OpenGL API published by SGI®, has some key differences and benefits. The technology used in SimGL can help application developers by providing a highly optimized API specifically designed for realtime 3D graphics. In general, SimGL can be considered a subset of OpenGL specifically designed and optimized for interactive 3D graphics applications.

SimGL's history stems from the port of Quantum3D's OpenGVS™ scene management software to the 3dfx® Glide graphics API. Since OpenGVS is based on OpenGL, the port to a 3dfx platform drove the development of an abstraction layer of software called Subset GL or SGL. In 1997, the lack of OpenGL drivers for the 3dfx Voodoo and Voodoo2 chipsets led to agreements between Quantum3D and 3dfx that resulted in a teamed effort of to form the product SimGL.

What is OpenGL?

Starting with an excerpt from http://www.OpenGL.org defines OpenGL as follows: "OpenGL fosters innovation and speeds application development by incorporating a broad set of rendering, texture mapping, special effects, and other powerful visualization functions. Developers can leverage the power of OpenGL across all popular desktop and workstation platforms, ensuring wide application deployment." Although this is not a complete technical description of OpenGL, this is a good starting point for comparison.

With this said it can be assumed that the OpenGL API in general is not specifically designed for the development of Realtime3D applications. Rather, OpenGL is primarily a robust mechanism for programming any 3D Graphics application, which would include CAD/CAM, Games, Realtime3D and more. In fact developers using OpenGL to develop a Realtime3D application must be very cautious not to use OpenGL functionality that may have an adverse affect on realtime performance on the target platform. Calling the wrong function can result in an intense amount of CPU or graphics computation that will likely severely impact an application's performance. A generic OpenGL compliant driver does not alone guarantee realtime performance, in fact it could imply otherwise.

What is SimGL?

SimGL is an API that uses the same calling conventions as OpenGL. In fact the function calls for SimGL are identical to OpenGL including arguments and function names. SimGL does not implement a compliant OpenGL driver. This means that not all functions are implemented and not all computations performed by the API are done in a compliant manner. SimGL has two native graphics APIs as its foundation; Direct3D and Glide. SimGL is a very fast API and it is not only fast on 3dfx graphics accelerators —the Direct3D implementation is fast on all PC graphics accelerators.

SimGL is typically faster than OpenGL because of the internal optimization done specifically for realtime 3D operations. In most cases using SimGL rather then OpenGL display drivers will result in increased performance, regardless of the accelerator being used. Display List operation performs extremely well in the SimGL API. SimGL is authenticated on Quantum3D Obsidian® and Obsidian®2 Professional graphics subsystems and does not require a license for Quantum3D Professional graphics subsystems and systems products.

SimGL Use of Low level Graphics APIs

SimGL has implementations that use Microsoft Direct3D and 3dfx Glide graphics APIs. Both Glide 2.X and Glide 3.X are supported by SimGL. SimGL DLLs for the Direct3D and Glide versions are available with all Quantum3D Professional Drivers. Versions of the GLU libraries are also available for applications developers that link directly with SimGL and also use GLU.

Developing an Application with SimGL

Developers must have access to the SimGL SDK. This SDK can be delivered as part of the OpenGVS SDK or as a standalone SimGL SDK. Either SDK can be obtained from a Quantum3D sales

SimGL䇊 And OpenGL®

How SimGL Optimizes OpenGL for Interactive 3D Graphics Applications

representative. OpenGL applications must be linked against SimGL. This enables the application to run in an OpenGL or SimGL runtime environment. It is possible to use OpenGL and SimGL simultaneously in the application if the application loads either the OpenGL DLL or SimGL DLL.

Running with SimGL

If the application has been linked against a SimGL library, then all that needs to be done is to place the desired SimGL library in the current working directory, a directory in the application's path or in the system directory. The same must be done for the GLU utility libraries if they are also being used. If the application has been linked against the OpenGL library, place SimGL in the directory along with the application or in the system directory. More details are supplied with the SimGL SDK documentation.

How SimGL works on Obsidian2 Graphics

Quantum3D Obsidian and Obsidian2 graphics subsystems support full screen 3D only. These accelerators are secondary display devices in Windows® 9X and Windows NT®. The application program's OpenGL window will appear on the Windows desktop as normal, but will appear blank. The SimGL 3D graphics output will appear on the secondary display and will be positioned in the lower left corner. The SimGL view port will be the same size, measured in pixels as the OpenGL window on the primary display. If a single display and the pass through feature of the Obsidian2 is used, only the SimGL 3D view port will be viewable. SimGL has been optimized for Display List rendering so even though immediate mode is fast it is not as efficient.

OpenGL Functions available in SimGL

Some functions in SimGL offer only a partial implementation of the OpenGL functionality. Many functions are defined only as stubs and issue a warning message the first time they are called. For a complete for a complete up-to-date list of these functions see the appendices of the SimGL

document *q3dsimgl.txt* that is supplied with the SimGL SDK.

SimGL Extension Functions

SimGL implements certain extensions that enable users to take advantage of unique features of Quantum3D Obsidian and Obsidian2 hardware. Extensions are also available for multi-channel support. This enables the development of PC systems that utilize more than one graphics accelerator per system. Other extensions are available to perform extremely efficient clipping and lighting. Other SimGL hint functions are available to avoid certain limitations of the range of texture coordinates on polygon vertices.

Conclusion

SimGL is a high performance API designed specifically for Realtime3D application development. It is a more efficient API that gives you improved performance over standard (compliant) OpenGL drivers. Because of SimGL's history and background with OpenGVS, it supports and is optimized for the specific functions needed by simulation and deployment applications. SimGL is a Realtime3D API used for application developers that require efficiency and high performance graphics. For more technical information on SimGL, please contact support@quantum3d.com.