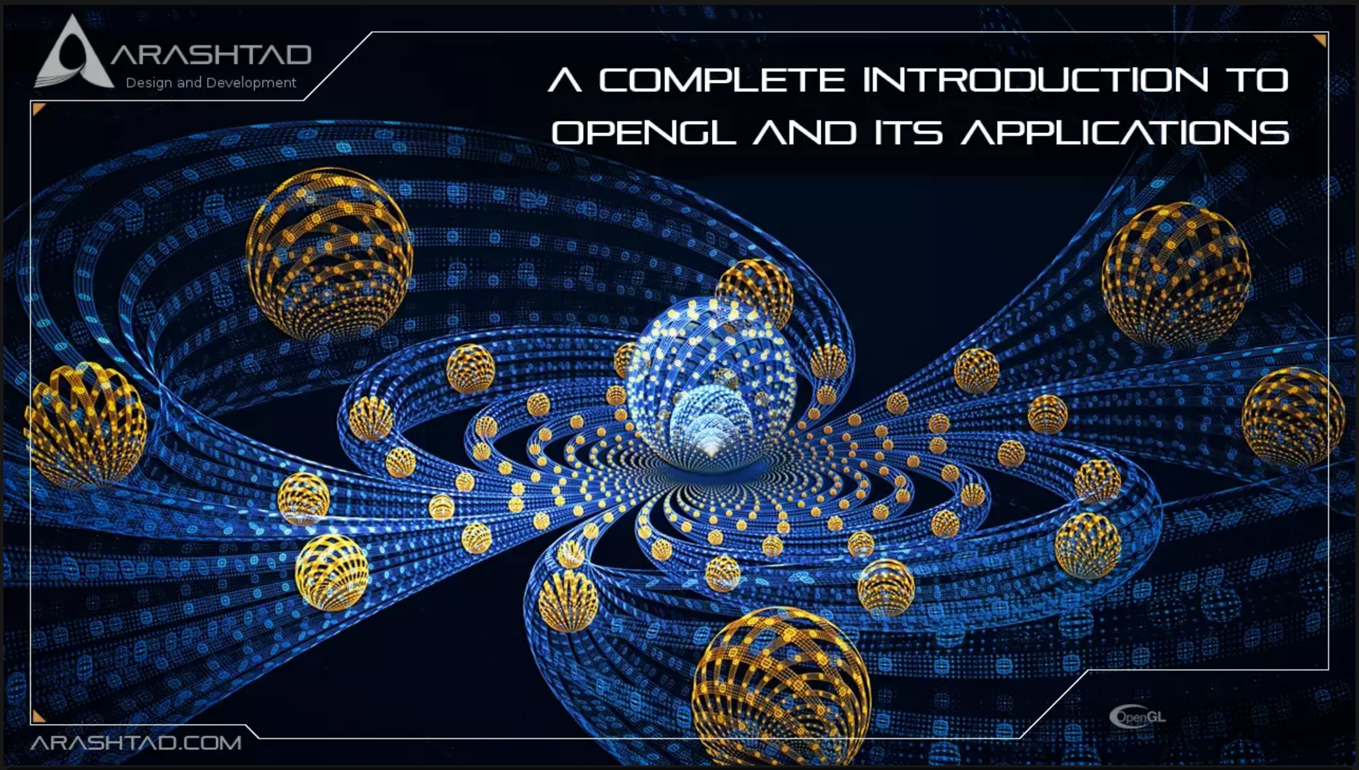
A Complete Introduction to OpenGL And its Applications

No comments



An open-source system for rendering vector graphics in 2D and 3D, OpenGL is a cross-platform and cross-language API. To achieve hardware-accelerated rendering, application developers typically use the API to communicate with graphics the processing unit (GPU).

What is OpenGL exactly?

In computing, OpenGL stands for Open Graphics Library. OpenGL is an API or standard specification that facilitates communication with graphic hardware (GPU). The OpenGL API provides a graphical representation of something in an application, similar to using the FS module API when accessing files in Node.js. With OpenGL, 2D, and 3D vector graphics can be rendered through hardware acceleration using GPUs, so it is widely used in VR, CAD, games, etc.

The Design of the OpenGL Graphics Interface

In OpenGL, two-dimensional and three-dimensional graphics are drawn using an abstract API. Even though the API can be implemented entirely in software, it is typically implemented mostly in hardware. APIs consist of a set of functions a client program can call along with named integer constants (e.g., GL_TEXTURE_2D, which has the decimal number 3553). Despite the superficial similarity, the definitions of the functions are independent of the programming language C.

Thus, OpenGL has many language bindings, with the JavaScript binding WebGL being particularly noteworthy (API based on OpenGL ES 2.0, which can render 3D images within a browser); the C bindings WGL, GLX, and CGL; iOS's C binding; the Java and C bindings provided by Android are also included. Additionally, OpenGL is cross-platform and language-independent. There is no mention in the specification of how to obtain and manage an OpenGL context, leaving it up to the windowing system. Likewise, OpenGL provides no APIs for inputs, audio, or windowing, since it is purely concerned with rendering.

OpenGL Development

Khronos Group releases new versions of OpenGL specifications each extending the API to include new features. In every version, graphic card manufacturers, operating system designers, and general technology companies vote for consensus. The vendors of graphics processing units (GPUs) may provide additional functionality as extensions in addition to the core API. Extensions may add new functions, and new constants, and may relax restrictions on existing OpenGL functions. Using extensions, vendors can expose custom APIs without having to seek support from other vendors or the Khronos Group as a whole, greatly increasing the flexibility of OpenGL. All extensions are defined in the OpenGL Registry.

The extension name, the constant, and the function associated with each extension are based on the name of the company that developed it. Nvidia, for example, is represented by NV in the extension name GL_NV_half_float, in the constant GL_HALF_FLOAT_NV, and in the function glVertex2hNV(). It is possible to release a shared extension, using the identifier EXT, if multiple vendors agree to implement the same functionality using the same API. In such cases, it is also possible the Khronos Group's Architecture Review Board will give their explicit approval to the extension, in which case the identifier ARB will be used. Typically, new versions of OpenGL introduce features derived from several widely-implemented extensions, especially ARB and EXT extensions.

What are the advantages of Open GL?

1. OpenGL is the only open, vendor-neutral, and multi-platform drawing standard. It is governed by an independent committee, the OpenGL Architecture Review Board. OpenGL has been approved by the Free Software Foundation in 2001. B. OpenGL is the only open, vendor-neutral, and multi-platform drawing standard.

2. Over the past seven years, OpenGL has been stable across a broad range of platforms. Updates to the specification are well-controlled, and proposed changes are announced in advance so that developers can

adopt them. Existing applications will not become obsolete due to backward compatibility requirements.

3. There is no difference between OpenGL applications and non-OpenGL applications in terms of reliability and consistency of visual display results, regardless of the operating system or windowing system:

   All Unix Workstations

   - Linux
   - BeOS
   - Mac OS
   - OpenStep
   - OS/2

   Python (mainly for data science and machine learning)
   Windows 95/98/NT/2000

4. Windows systems support the following:
   - Presentation Manager
   - Win32
   - X/Window System

5. Several languages are available for developers to access OpenGL applications and services, including:
   - Ada
   - C
   - C++
   - Fortran
   - Java

6. The OpenGL software is independent of all network protocols and computer architectures.

7. OpenGL is scalable with all applications that run on a wide variety of hardware, from consumer electronics to PCs, workstations, and supercomputers. Therefore, developers are free to target any type of machine they wish.

WebGL vs OpenGL | Know The 7 Most Useful Differences

Here are some of the major differences between these two popular choices:

1. WebGL does not support some OpenGL features, including geometry shaders, tessellation shaders, and compute shaders.

2. WebGL uses OpenGL ES, which lacks some of the features found in regular OpenGL, such as vertex and fragment shaders.
3. WebGL is primarily used for browsers. OpenGL does need native drivers and is primarily used for installing software.
   Several video games use OpenGL, while web applications use WebGL.
4. Learning WebGL and developing applications is easier. Learning OpenGL is easier if you know WebGL.
5. WebGL is capable of faking 3D textures using a 2D texture, but OpenGL doesn't require this since it has more features such as geometry and shaders.
6. Shaders and buffers are forced to be learned from the start in WebGL, but not in OpenGL.
7. OpenGL has a bigger learning curve since it has a lot of features, including WebGL, whereas WebGL has a smaller learning curve due to its fewer features.
8. OpenGL ES is a subset of OpenGL. It has fewer capabilities and is easier for users to use. OpenGL has many capabilities. WebGL takes advantage of OpenGL's advantages.

Both OpenGL and WebGL are graphics libraries for rendering two-dimensional and three-dimensional images. WebGL is being used in the HTML canvas element, which means it is a part of the HTML language. WebGL is easy to learn because it is simple and easy to use in JavaScript and HTML. OpenGL requires good knowledge to work with and develop the application.

There are advantages and disadvantages to both OpenGL and WebGL. The selection of graphics libraries depends on the requirements, scalability, and future vision of the application. To develop and upgrade one's skill set, it is always important to learn, explore, and use different technologies.

Conclusion

The purpose of this article is to introduce you to OpenGL, its design, its development, and its advantages. A cross-platform, hardware-accelerated, language-independent, industrial standard API for producing 3D (including 2D) graphics, OpenGL (Open Graphics Library) is an open-source, cross-platform, hardware-accelerated API. Modern computers are equipped with a GPU with dedicated memory to speed up graphics rendering. A graphics card interface is OpenGL. In other words, your application could issue OpenGL graphics rendering commands to the graphics hardware, which accelerated them.

Join Arashtad Community

Follow Arashtad on Social Media

We provide variety of content, products, services, tools, tutorials, etc. Each social profile according to its features and purpose can cover only one or few parts of our updates. We can not upload our videos on SoundCloud or provide our eBooks on Youtube. So, for not missing any high quality original content that we provide on various social networks, make sure you follow us on as many social networks as you're active in. You can find out Arashtad's profiles on different social media services.

Get Even Closer!

Did you know that only one universal Arashtad account makes you able to log into all Arashtad network at once? Creating an Arashtad account is free. Why not to try it? Also, we have regular updates on our newsletter and feed entries. Use all these benefitial free features to get more involved with the community and enjoy the many products, services, tools, tutorials, etc. that we provide frequently.

SIGN UP NEWSLETTER RSS FEED