
DIALOGUES WITH SPECIALISTS

Review article

UDC 681.3; DOI: 10.61260/2304-0130-2024-2-82-87

GRAPHICAL TOOLS OF OPERATING SYSTEMS

Labinskij Aleksandr Yu.

Saint-Petersburg university of State fire service of EMERCOM of Russia, Saint-Petersburg, Russia
Labinsciy@yandex.ru

Abstract. The methods of processing multimedia and graphical information in operating systems are studied using the example of MCI, API and GDI interfaces of Windows operating system. The interfaces of GDI and GDI+ graphics devices are examined in detail. The capabilities and features of the OpenGL graphics library, which is commonly used in the Windows operating system, are presented. The principles of operation of the GDI interface are given. Three types of drawing tools supported by the context of the GDI interface are examined in detail: pen, brush and font. The stages of coordination of logical and physical coordinate systems are considered. The OpenGL library graphics elements (bits arrays, polygons, segments, polylines, etc.), as well as basic geometric objects and a set of polyhedra: cube, cone, torus, octahedron, icosahedron, dodecahedron and tetrahedron are considered. As an example of OpenGL library usage, a three-dimensional graphics editor was studied. It works with three-dimensional objects of various types and allows reproduction of an object in two modes (lines and a solid surface), as well as performance of following operations with a graphic object: deformation, rotation, transposition along axes, loading and saving on hard drive.

Keywords: graphical tools, Linux operating system, Windows operating system, graphical information, API interfaces, GDI and MCI, GDI and GDI+ graphics device interfaces, OpenGL graphics library, three-dimensional graphics editor

For citation: Labinsky A.Yu. Graphical tools of operating systems // Supervisory activities and forensic examination in the security system. 2024. № 2. C. 82–87. DOI: 10.61260/2304-0130-2024-2-82-87.

Introduction

For mass use of technical training tools (visualizers, interactive whiteboards, installers, etc.) in the educational process, it is necessary to develop the necessary methodological materials that can be used on the basis of computer graphics with the capabilities described in [1–11].

Computer graphics have extensive capabilities, including not only acceleration of information perception and intuition development, but also increase of students' cognitive abilities, which are based on imaginative and logical thinking.

Many operating systems (OS) have integrated graphical tools, including various versions of Linux and Windows OS. Linux OS distributions, in particular Astra Linux, are involved in implementation of transition program of federal budget institutions to the use of independent software [1]. Though, Windows OS is still used on many personal computers in Russia. The capabilities of computer graphics on Windows OS are discussed in this article.

Windows OS allows one to set various graphics modes, including 16-colour mode, 256-colour mode, 65000-colour mode and some others. The main task of Windows computer graphics is image construction, which is based on the creation of a mathematical model of the depicted object and on the visualization of the object in accordance with this model.

Windows OS also allows the use of OpenGL and DirectX graphics libraries [2], which offer a graphical interface between software and hardware. The OpenGL and DirectX libraries implement many operations on flat and three-dimensional objects: rotation, resizing and illumination, scale transformations, etc.

The purpose of this article is to review the graphical tools of Windows OS. The topic of the article is relevant, since the use of graphic information in the learning process increases the cognitive abilities of students based on imaginative and logical thinking, as well as accelerates the development of intuition. All this helps to increase the level of understanding of the studied disciplines, which can lead to an increase in the academic performance.

The novelty of the research lies in a detailed examination of the API application programming interfaces, MCI multimedia and GDI and GDI+ graphics devices, as well as the capabilities of the OpenGL graphics library. In addition, the three-dimensional graphics editor is considered as an example of using the OpenGL library.

Graphical tools of Windows OS

Windows OS uses API (Application Program Interface), GDI (Graphics Device Interface) and MCI (Media Control Interface) for processing graphical and multimedia information.

The API interface contains a set of functions implemented by the operating system in order to provide services to executable applications, interact between them, and control computer devices from applications. Support for such functions at the OS level ensures that there is no need to program such functions in each application and implements the interaction of applications with the operating system.

The GDI interface supports special drawing techniques and contains various graphic classes:

- the «Graphics Object» class, which includes the main classes of graphical tools;
- the «Canvas» class, which contains drawing tools and methods;
- the «Graphics» class, which generates four classes of images with its own format: a bits field, icons, metafiles and images in the *.JPEG format;
- the «Picture» class, which contains a drawing field (canvas), which can include a field of bits, icons, metafiles and images in the *.JPEG format.

The MCI interface supports the operation of the following main devices:

- optical CD player;
- DVD player;
- audio output devices in MIDI format;
- audio output devices in WAV format;
- AVI output devices.

The OpenGL Graphics Library (Open Graphics Library) contains hundreds of procedures and functions that allow one to describe three-dimensional graphic objects and operations on them [3]. These procedures and functions allow one to:

- draw geometric and raster primitives;
- work with color in RGB mode and in index mode;
- implement Visual transformations and set the lighting model;
- remove invisible edges and faces on the objects being drawn;
- use various effects, including color pairing, transparency, texture overlay, lighting with distributed and point light sources, the «fog» effect and many others;
- use line smoothing using splines and Bezier curves.

Graphic devices interface

The structure of the graphics device interface (GDI), which provides interaction between the application program and the graphics output devices, is shown in fig. 1.

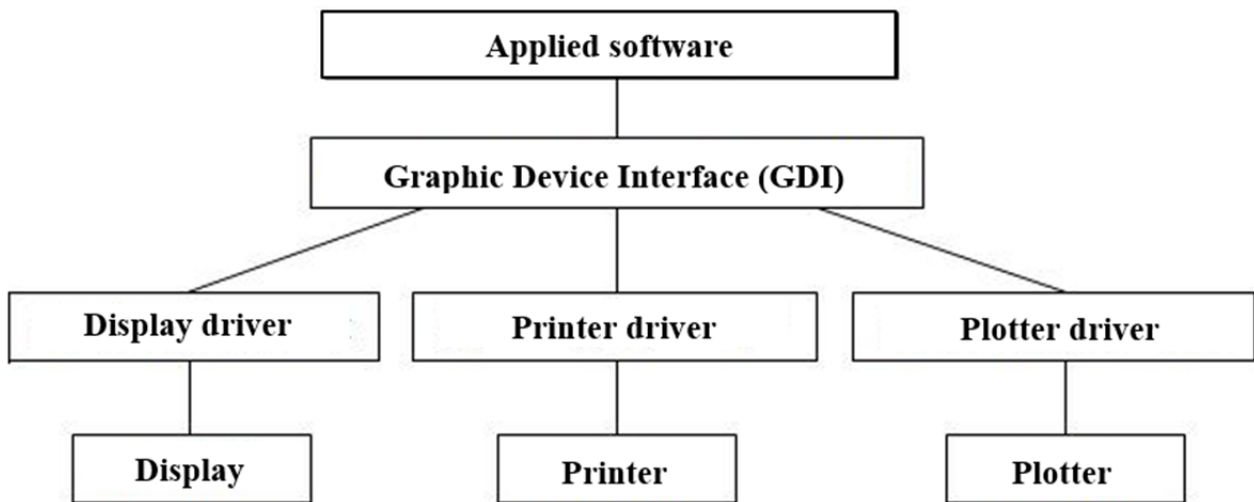


Fig. 1. **Application program interaction with graphic output devices**

The basis of GDI's framework [4]:

- the GDI interface is hardware independent and treats any graphics device as a virtual output device;
- the GDI interface uses special programs (drivers) that convert the output values of GDI functions into commands executed by output devices;
- in GDI, to describe geometric objects and their relative positions in space, a coordinate system is introduced and each point in the graphic space has a corresponding set of real numbers – the coordinates of this point;
- the number of coordinates in a set of numbers describing a point in space determines the dimension of the graphic space;
- geometric objects in graphic space are described using Cartesian, cylindrical and spherical coordinate systems;
- three coordinate systems are used in computer graphics: a fixed world system, an object coordinate system associated with a specific object, and an on-screen coordinate system associated with a graphical output device (display, etc.).

The display on which the information is Output is a shared device. The display is divided between applications using output windows. Windows OC speeds up graphical output to the *output window* by optimizing the redrawing of the graphical object by clipping areas that do not require updating.

The GDI interface with graphics devices works based on the concept of a device context (DC – Device Context), which is a logical object of Windows OS, is associated with a physical output device and replaces it in graphics output functions. The Device Context (DC) supports three types of drawing tools: pen, brush, and font. GDI always uses a pen, brush, and font connected to the device context to which the output is routed.

The pen is a graphical tool for drawing lines and contours of various geometric shapes and is characterized by style, thickness and color.

The brush is a graphical tool for painting over various areas of the image and is characterized by a type of painting: solid color, hatching or repetition of the image (pattern).

A font is a set of characters with similar sizes and outlines.

The physical coordinate system uses pixels as units of measurement. A logical coordinate system can use pixels, millimeters, or inches as units of measurement.

The coordination of the logical and physical coordinate systems is carried out by recalculating the coordinates of the graphic object, which is performed in three stages:

- the stage of parallel image transfer to a logical plane by subtracting the set constant values of the coordinates of each point of the image ;

- scaling stage – compression (stretching) of the resulting image by multiplying the coordinates of each point of the image by the specified scale coefficients;
- the stage of parallel image transfer to the physical plane by adding the specified constant values to the coordinates of each image point.

The GDI interface provides applications not only with means of displaying graphics, but also with means of storing graphical information. Graphic information can be stored in BMP, GIF, ICO, WMF files.

Windows metafiles (WMF) are one of the forms of storing graphic images. The image recorded in the metafile is, unlike a bitmap image, a sequence of commands from the GDI interface. The advantage of GDI commands is that they provide image playback on any graphics device, since the metafile stores the numbers of commands (functions) of the GDI interface along with the values of their parameters.

Basics of the OpenGL Graphics Library

The OpenGL library was developed in the late XX – early XXI centuries by leading companies such as Apple, Dell, Hewlett-Packard, IBM, Microsoft, SGI, Sun Microsystems. The basis was the IRIS GL graphics library, developed by SGI (Silicon Graphics Incorporated). The OpenGL library is based on dynamic libraries contained in files OpenGL32.dll and glu32.dll [5].

The OpenGL library commands include an interpretation mechanism involving the use of client-server technology. This technology is based on such a mechanism for transmitting commands from the client program to the server program, according to which the server program can be located on the same or on another computer.

The OpenGL library contains many hundreds of functions and procedures that provide the ability to describe three-dimensional graphical objects and perform operations on them. These procedures and functions allow [1]:

- output geometric and raster primitives;
- use color in RGB mode and in index mode;
- use transformations of viewing angles of graphic objects and modifications of lighting models;
- to cut off invisible edges and faces on the objects being drawn;
- use various effects, including color pairing, transparency modification, texture overlay, lighting with distributed and point light sources, the «fog» effect and many others;
- Use line smoothing using splines and Bezier curves.

The OpenGL library contains the following graphic primitives [6]: bits arrays, line segments, polylines, icons, metafiles and images in *.JPEG format.

The OpenGL library has basic geometric objects [2]: points arrays, polylines and closed lines, polygons of various types and connected to each other in various ways.

In the DGLUT module of the OpenGL library (file glu32.dll) a set of polyhedra is presented: cube, cone, torus, octahedron, icosahedron, dodecahedron and tetrahedron. Polyhedra can be displayed in two ways: by rendering a wireframe and by solid representation.

In the OpenGL library, view transformations are performed using a view matrix, a projection matrix, and a texture matrix. Special commands are used for the operations of transferring, rotating and scaling the coordinate system. Objects are projected using perspective and parallel projections.

In the OpenGL library, each object has a material with the properties of Scattered and diffuse colors, specular reflection colors, and radiated colors.

The OpenGL library supports the overlay of one-dimensional and two-dimensional images on the surface of a graphic object.

3D Graphics Editors

The graphic editor shown in fig. 2 [6] is based on the standard Windows graphics library – OpenGL and works with objects of nine types: parallelepiped, sphere, cylinder, cone, torus, dodecahedron, icosahedron, tetrahedron and the standard object of editors of this type – kettle.

In the right part of the graphical editor window there is a control panel with three buttons: the button for reading the projected model («Load»), the button for recording the projected model («Save») and the button for redrawing the screen («Refresh»).

At the bottom of the panel there are several components that specify the presence of axes, a grid, fog and a platform providing orientation in space in the graphical output area.

A graphic object is reproduced in two modes: lines or a solid surface. A graphic object can be deformed, rotated and shifted along the X, Y, Z axes, loaded and saved to disk.

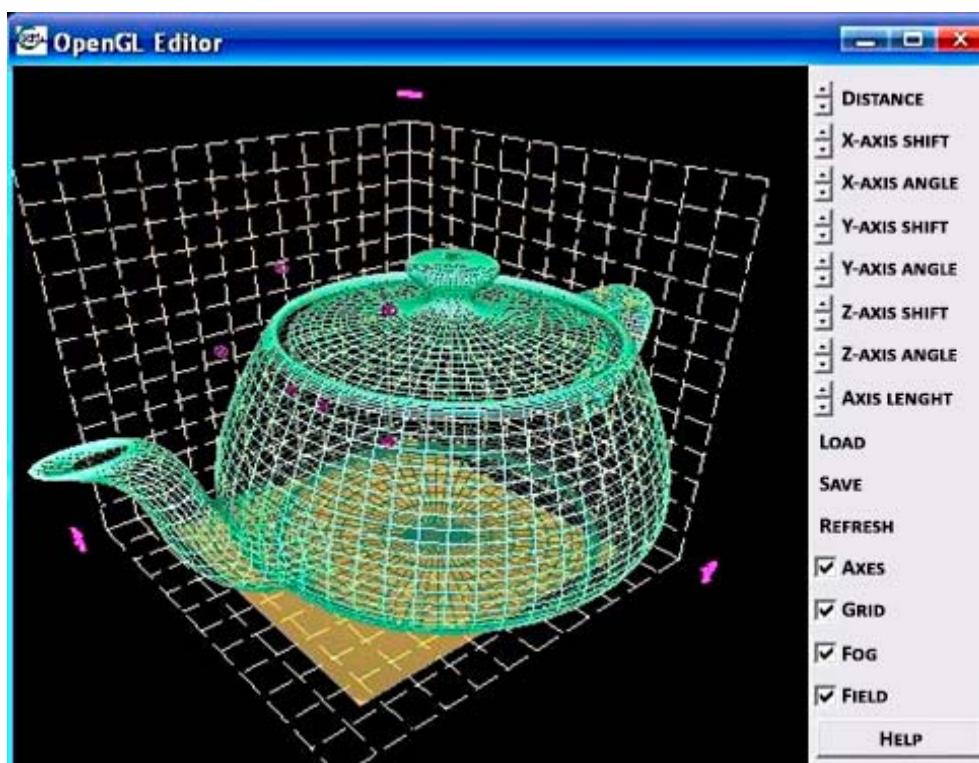


Fig. 2. Three-dimensional graphics editor

Conclusion

Linux OS distributions, in particular Astra Linux, are involved in implementation of transition program of federal budget institutions to the use of independent software. However, Windows OS is still used on many personal computers in Russia, and there is a lot of experience in using the graphics tools of this OS, which can be used, for example, in the Astra Linux OS.

The Windows API, GDI, GDI+ and MPI interfaces that implement methods for processing graphic and multimedia information are considered in detail.

The possibilities and features of the OpenGL graphics library, which is actively used in the Windows operating system environment, are presented.

As an example of using the OpenGL library, a three-dimensional graphics editor was examined. It works with three-dimensional objects of various types and allows you to reproduce an object in two modes (lines and a solid surface), as well as perform the following operations with a graphic object: deform, rotate, shift along axes, load and save to disk.

References

1. Labinsky A.Yu. Features of astra linux operating system // Supervisory activities and forensic examination in the security system. 2024. № 1. P. 43–48.
2. Adams D. DirectX: advanced animation. M.: Kudic, 2004. 287 p.
3. Krasnov M.V. OpenGL. Graphics in Delphi projects. SPb.: BHV-Peterburg, 2006. 352 p.
4. Tihomirov Yu.V. Programming of three-dimensional graphics. SPb.: BHV-Peterburg, 2007. 256 p.
5. Evchenko A. OpenGL и DirectX. Graphics programming. SPb.: Piter, 2008. 352 p.
6. Nikulin E. Computer geometry and machine graphics algorithms. SPb.: BHV-Peterburg, 2009. 288 p.
7. Yuang Feng. Graphics programming for Windows. SPb.: Piter, 2012. 368 c.
8. Clarc J.H. OpenGL Library for Graphic. London, 2006.
9. Hans J., Max M. Computer Graphic Hardware. Hamburg: FRG, 2009.
10. Adrian N. Computer-Generated Pictures. Viena, 2011.
11. Weiler K. Introduction to Solid Modeling. N.Y., 2001.

Information about the article: the article was received by the editor: 26.04.2024;
accepted for publication: 15.05.2024

Information about the authors

Labinsky Alexander Yu., associate professor of the department of applied mathematics and information technologies of the Saint-Petersburg university of State fire service of EMERCOM of Russia (196105, Saint-Petersburg, Moskovsky ave., 149), PhD in technical sciences, associate professor, e-mail: Labinsciy@yandex.ru, <https://orcid.org/0000-0001-2735-4189>, SPIN: 8338-4230