



Facultad de Estudios Superiores

**Acatlán**

# Gráficos por computadora usando JOGL



**Oscar Gabriel  
Caballero Martínez**

# **Gráficos por Computadora Usando JOGL**



UNIVERSIDAD NACIONAL AUTÓNOMA DE  
MÉXICO

**Dr. Enrique Luis Graue Wiechers**

Rector

**Dr. Leonardo Lomelí Vanegas**

Secretario General

FACULTAD DE ESTUDIOS SUPERIORES  
ACATLÁN

**Dr. Manuel Martínez Justo**

Director

**Mtra. Nora del Consuelo Goris Mayans**

Secretaria General

**Mtro. Carlos Nandayapa Hernández**

Secretario de Estudios Profesionales

**Dra. Laura Páez Díaz de León**

Secretaria de Posgrado e Investigación

**Mtro. Fernando Martínez Ramírez**

Coordinador de Servicios Académicos

**Act. Luz María Lavín Alanís**

Jefa de la División de Matemáticas e  
Ingeniería

**D.G. Norma Guadalupe Rojas Borja**

Jefa de la Unidad de Servicios Editoriales



Facultad de Estudios Superiores

**Acatlán**

# Gráficos por computadora usando JOGL

**Oscar Gabriel  
Caballero Martínez**



**Catalogación en la publicación UNAM.  
Dirección General de Bibliotecas y  
Servicios Digitales de Información**

**Nombres:** Caballero Martínez, Oscar Gabriel, autor.

**Título:** Gráficos por computadora usando JOGL / Oscar Gabriel Caballero Martínez.

**Descripción:** Primera edición. | México: Universidad Nacional Autónoma de México, Facultad de Estudios Superiores Acatlán, 2021.

**Identificadores:** LIBRUNAM 2112231 (libro electrónico) | ISBN 9786073050302 (libro electrónico).

**Temas:** Gráficas por computadora. | OpenGL. | Java (Lenguaje de programación para computadora). | Imágenes tridimensionales.

**Clasificación:** LCC T386.064 (libro electrónico) | DDC 006.6—dc23

Primera edición digital: 2021

D.R. © UNIVERSIDAD NACIONAL  
AUTÓNOMA DE MÉXICO  
Ciudad Universitaria, Alcaldía  
Coyoacán,  
C.P. 04510, Ciudad de México, México.

FACULTAD DE ESTUDIOS SUPERIORES  
ACATLÁN  
Av. Alcanfores y San Juan Totoltepec  
s/n  
C.P. 53150, Naucalpan de Juárez,  
Estado de México.  
Unidad de Servicios Editoriales.

*Esta edición y sus características son  
propiedad  
de la Universidad Nacional Autónoma de  
México.*

*Prohibida la reproducción total o parcial  
por cualquier medio sin la autorización  
escrita  
del titular de los derechos patrimoniales.*

ISBN: 978-607-30-5030-2

Hecho en México  
*Made in Mexico*

# Índice

[Introducción](#)

[Capítulo 1 Principios de gráficos por computadora](#)

[Capítulo 2 Programación con Java Open Graphics Library 2. \(JOGL2\)](#)

[Capítulo 3 Primitivas Gráficas](#)

[Capítulo 4 Algoritmos para generar líneas](#)

[Capítulo 5 Transformaciones afines en 2D: traslación, escalamiento, rotación](#)

[Capítulo 6 Gráficas en 3D](#)

[Capítulo 7 Aplicación de los modos de vista](#)

[Capítulo 8 Gráficas de curvas y superficies](#)

[Capítulo 9 Técnicas de barrido e iluminación](#)

[Capítulo 10 Animación y juegos](#)

[Referencias](#)

[Apéndice A](#)

[Apéndice B](#)

# Introducción

Este documento está dirigido a personas interesadas en la generación de gráficos por computadora con el lenguaje de programación Java.

A lo largo del texto se descubrirá que la mayoría de los trabajos que existen para la generación de gráficos por computadora están basados en el uso de las librerías conocidas como OpenGL. Las referencias más accesibles están trabajadas con el lenguaje de programación C, lenguaje principal en el que se han creado las funciones de la librería.

El reto consiste en presentar el uso de estas librerías con el lenguaje de programación Java, de ahí las siglas JOGL (Java Open Graphics Library o Java OpenGL).

La importancia del presente trabajo radica en mostrar la facilidad con la cual los programas que se encuentran en las referencias del presente documento también pueden ser escritos en el lenguaje Java, y así poseer un panorama más amplio que ayude a la profundización del conocimiento para la creación de las gráficas por computadora.

El estudio de las librerías de OpenGL permite, a quienes las trabajan, entender mejor la terminología y funciones que se utilizan en la generación de gráficos por computadora por medio de diferentes paquetes que existen en el mercado, como Maya y 3D Studio Max, entre otros, y comprender los rudimentos del manejo de lenguajes como WebGL.

El trabajo presenta una breve cronología de las máquinas y eventos que han generado el estudio de esta disciplina.

También trata sobre las principales funciones para facilitar la creación de figuras básicas relacionadas con el desarrollo de proyectos referidos a las gráficas computacionales, los efectos que pueden aplicarse con los objetos creados para una escena y las maneras en que se manipulan los elementos que intervienen en tales escenas, como es la iluminación, la animación y el sombreado.

Es importante destacar que el trabajo está enfocado a personas que ya han programado con el lenguaje Java, ya que el estudio no se detiene a verificar cuestiones relacionadas con el lenguaje, al igual que al manejo de la API NetBeans.

Sin más preámbulos, se invita al lector a realizar un recorrido por el maravilloso y fascinante mundo de la generación de gráficos por computadora mediante las librerías JOGL.

# Capítulo 1

## Principios de gráficos por computadora

Para tener un panorama más amplio sobre la historia de los **gráficos por computadora** a continuación se enlistan algunas fechas y eventos importantes en el desarrollo cronológico de la evolución de esta actividad.

### 1.1 Desarrollo cronológico

Durante la Segunda Guerra Mundial se realizó un proyecto de simulación de vuelo para entrenar pilotos. Se llevó a cabo en el MIT y estuvo a cargo de Jay Forrester

[Benstead, 2009]. Se concluyó hasta 1951. Llevó el nombre de *Whirl-wind*.



Figura 1.1 Computadora Whirl-wind

Fuente: [https://en.wikipedia.org/wiki/Whirlwind\\_I](https://en.wikipedia.org/wiki/Whirlwind_I)

En 1959, IBM y General Motors lograron modelar autos en 3D. Para 1962 Ivan Sutherland creó el Sketchpad, programa informático que permitía la manipulación directa de objetos gráficos: fue el primer programa de dibujo por computadora.

En 1963 Doug Englebart inventó el “*ratón*”, herramienta importante para la manipulación de gráficos en una

computadora, además de que ayuda a dibujar objetos con mayor facilidad. Para 1969 nace SIGGRAPH (Special Interest Group for Graphics), organismo que se encarga de estandarizar las técnicas para graficar en dispositivos electrónicos.

En el año de 1972 la Universidad de Utha desarrolla algoritmos y técnicas, como el sombreado de Goraud y Phong, el algoritmo del pintor y el algoritmo z-buffer, que son actualmente estudiados en los niveles básicos para el desarrollo de efectos visuales [Villar, 2009].

A principios de los 80's salen a la venta las computadoras Apollo, Macintosh e IRIS.

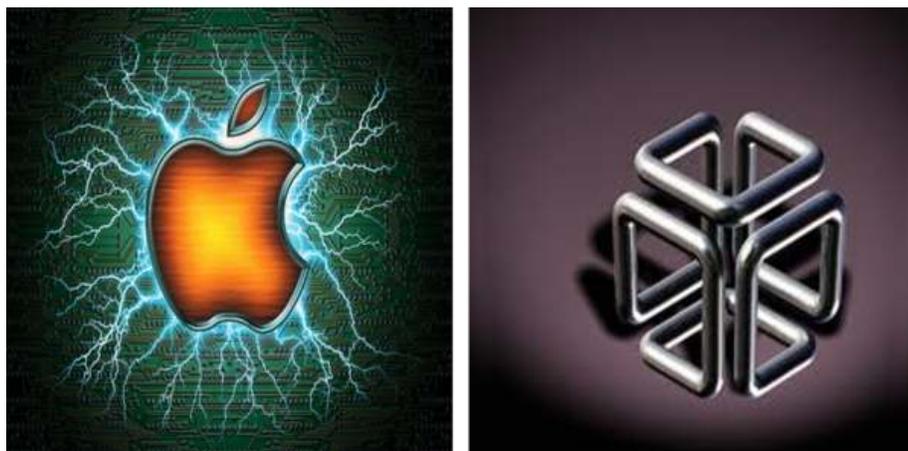


Figura 1.2 Macintosh y Silicongraphics  
Fuente: Macintosh.org & silicongraphics.org

En 1985 se crea un API estándar: GKS-  
> PHIGS

En el año de 1986 se realiza el primer cortometraje por computadora: *Luxo Jr.*



Figura 1.3 Luxo Jr.  
Fuente: [pixar.org](http://pixar.org).

Para 1990 aparece el primer producto de animación: Studio 3D de Autodesk.

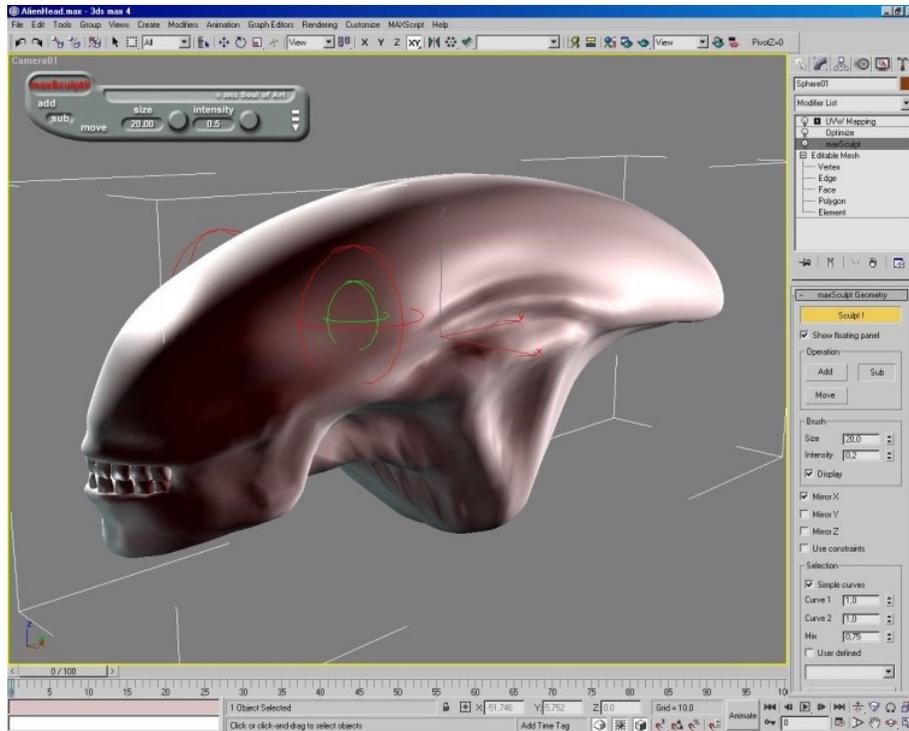


Figura 1.4 3D Studio Max.

Fuente: [www.fiuxi.net/autodesk-3d-studio-max-8-full-medicina.html](http://www.fiuxi.net/autodesk-3d-studio-max-8-full-medicina.html)

En 1995 se realiza la primera película animada por computadora: *Toy Story*.



Figura 1.5 Woody y Buzz

Fuente: [www.mundodisney.net/juegos/toy/009/](http://www.mundodisney.net/juegos/toy/009/)

Para generar una escena es importante seguir una serie de pasos para tener una mejor presentación.

## 1.2 principios generales

*Modelado:* Estudia los diversos tipos de modelos matemáticos usados para representar la forma de los objetos que se utilizan en una escena [Shreiner, 2013; Villar, 2009].

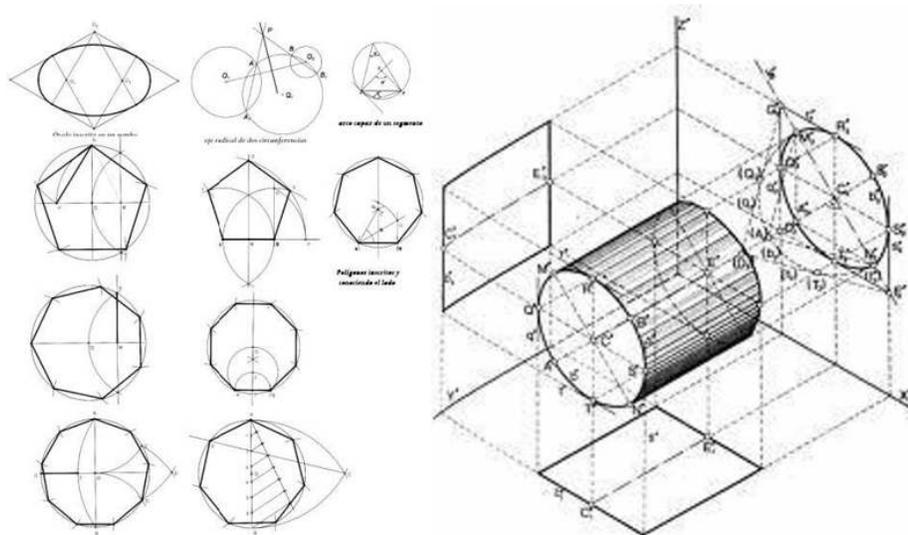


Figura 1.6 Modelado Geométrico

Fuente: Apuntes Villar Patiño.

*Render:* Es el proceso en el que se usan cálculos y algoritmos computacionales

para dibujar un objeto tridimensional de manera realista aplicando color, texturas y efectos de iluminación.

Para generar un *rendereo* primero se coloca el color a la imagen y posteriormente la iluminación.

La totalidad del proceso de render se presenta en la figura 1.7. A partir de figuras geométricas o una imagen se aplican los comandos para cada uno de los pixeles y por medio de las texturas permite una mejor representación de la figura, ya que al aplicarle a cada fragmento las operaciones pertinentes, podemos enviar al *frame* la figura para que sea dibujada [Hearn, 2006; Villar, 2009].

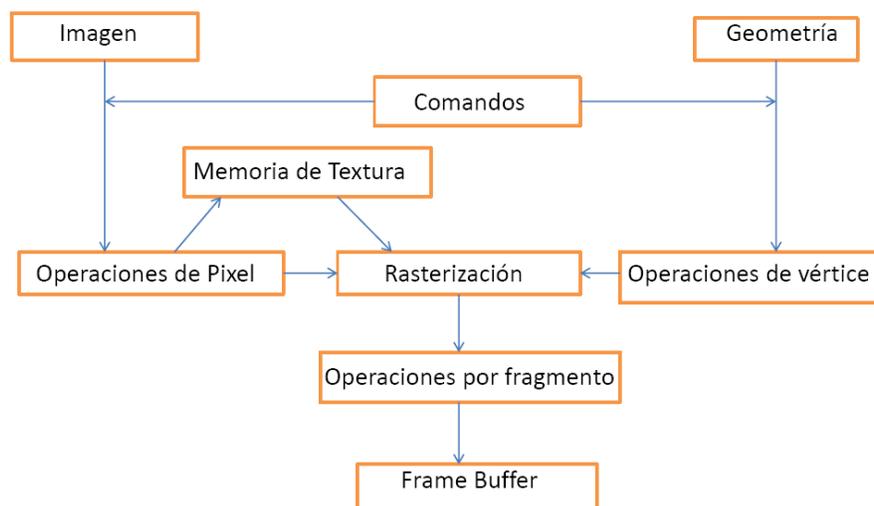


Figura 1.7 Pasos a seguir para el rendero.  
Fuente: Apuntes Villar Patiño.

*Animación:* Es el control computacional de una secuencia de imágenes u objetos a lo largo del tiempo

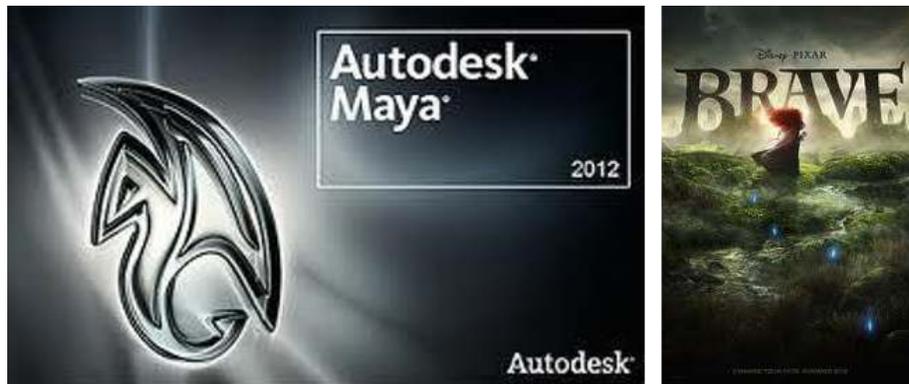


Figura 1.8 Maya Autodesk y Valiente de Pixar  
Fuente: [ielp1990blogspot.mx/2012/12/maya.html](http://ielp1990blogspot.mx/2012/12/maya.html) y [pixar.org](http://pixar.org)

*Interacción:* Estudia lo relativo a la creación, en hardware y software, de interfases de usuario.

- Para el hardware, se identifican los dispositivos que deben contener las computadoras para un mejor manejo.
  - Adaptador gráfico.
  - Monitores.
  - Dispositivos de entrada y salida.
- Para el software, se tienen las siguientes identificaciones que

permiten manejar la interacción de una mejor forma.

- GUI: Interfaz Gráfica del Usuario, por sus siglas en inglés.
- API: Conjunto de funciones y procedimientos que pueden ser usados por un programa para un fin específico.

# Capítulo 2

## Programación con Java

### Open Graphics Library 2. (JOGL2)

Para poder programar en JOGL2 [jogamp, 2016] es necesario tener las librerías que utiliza para su programación en Java (ver Apéndice A) y a su vez, configurar NetBeans para ligar las librerías a los proyectos que se crean (ver Apéndice B).

Lo primero que debe hacerse es generar, a partir del proyecto de trabajo, un archivo tipo *class* para transcribir el programa; para ello, dentro de NetBeans hacemos lo siguiente: con el botón derecho del mouse abrimos el menú en el

proyecto y en la opción de Java Class podremos construir el programa. Ver figura 2.1

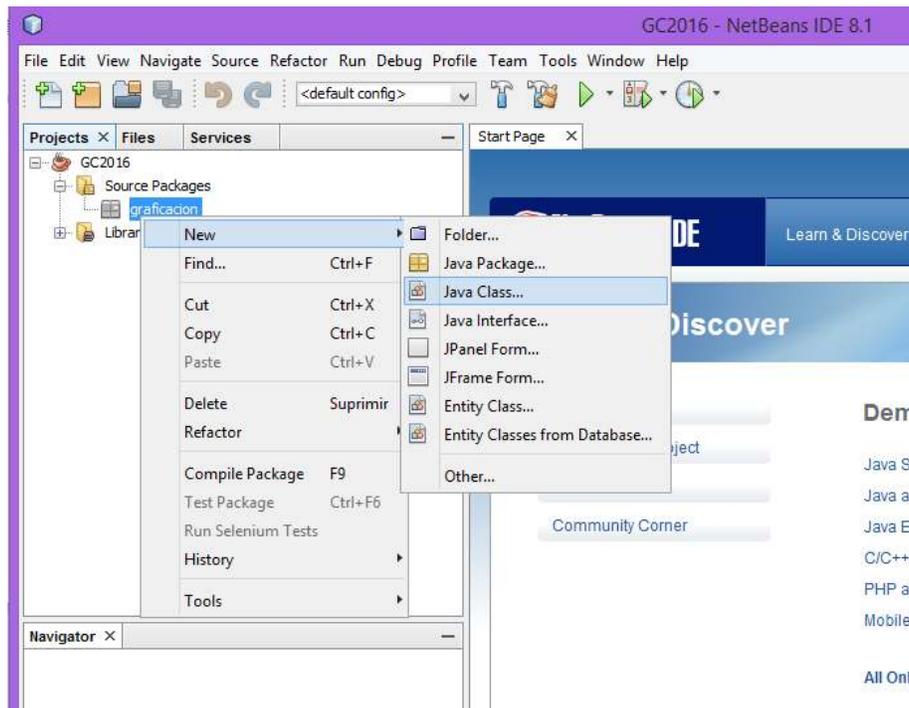


Figura 2.1 Ventana de NetBeans para crear un Java Class.  
Fuente: Elaboración propia

Al hacer lo anterior se pone el nombre de la clase en la ventana, como se muestra en la figura 2.2. Hay que recordar que los nombres de las clases en Java deben comenzar con una letra mayúscula; si el nombre que se desea transcribir es de dos o más palabras, cada nueva palabra comienza con Mayúsculas.

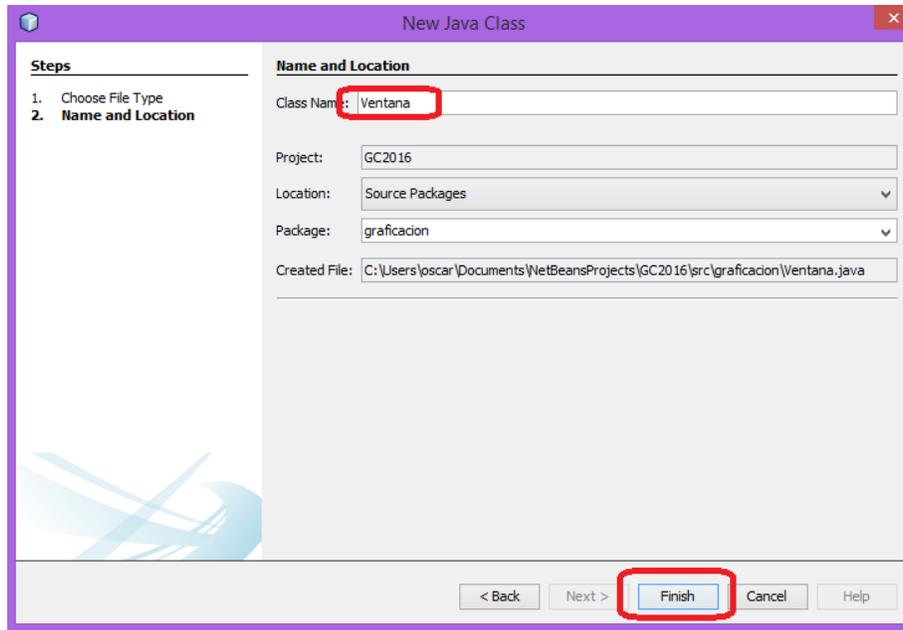


Figura 2.2 Poner nombre a una clase.  
Fuente: Elaboración propia

Al oprimir el botón de *finalizar* (*Finish*) se tendrá abierto el editor para transcribir programas en Java que genera un archivo \*.class. Para ver cómo se trabaja con JOGL se transcribirá el siguiente código [jogamp, 2016].

1. package Graficacion;
- 2.
3. import com.jogamp.opengl.GLAutoDrawable;
4. import com.jogamp.opengl.GLEventListener;
5. import com.jogamp.opengl.awt.GLCanvas;
6. import com.jogamp.opengl.util.Animator;
7. import java.awt.Frame;
8. import java.awt.event.WindowAdapter;
9. import java.awt.event.WindowEvent;