

Dpto. de Lenguajes y Sistemas Informáticos  
Escuela Técnica Superior de Ingenierías Informática y  
Telecomunicación

## **Prácticas de Informática Gráfica**

**Autores:**

Pedro Cano  
Antonio López  
Domingo Martín  
Juan carlos Torres  
Carlos Ureña

Curso 2013/14



## **La Informática Gráfica**

La gran ventaja de los gráficos por ordenador, la posibilidad de crear mundos virtuales sin ningún tipo de límite, excepto los propios de las capacidades humanas, es a su vez su gran inconveniente, ya que es necesario crear toda una serie de modelos o representaciones de todas las cosas que se pretenden obtener que sean tratables por el ordenador.

Así, es necesario crear modelos de los objetos, de la cámara, de la interacción de la luz (virtual) con los objetos, del movimiento, etc. A pesar de la dificultad y complejidad, los resultados obtenidos suelen compensar el esfuerzo.

Ese es el objetivo de estas prácticas: convertir la generación de gráficos mediante ordenador en una tarea satisfactoria, en el sentido de que sea algo que se hace “con ganas”.

Con todo, hemos intentado que la dificultad vaya apareciendo de una forma gradual y natural. Siguiendo una estructura incremental, en la cual cada práctica se basará en la realizada anteriormente, planteamos partir desde la primera práctica, que servirá para tomar un contacto inicial, y terminar generando un sistema de partículas con animación y detección de colisiones.

Esperamos que las prácticas propuestas alcancen los objetivos y que sirvan para enseñar los conceptos básicos de la Informática Gráfica, y si puede ser entreteniéndolo, mejor.



---

# Índice general

<b>Índice General</b>	<b>5</b>
<b>1. Introducción. Visualización de modelos PLY</b>	<b>7</b>
1.1. Objetivos . . . . .	7
1.2. Desarrollo . . . . .	7
1.3. Evaluación . . . . .	8
1.4. Extensiones . . . . .	8
1.5. Duración . . . . .	8
1.6. Bibliografía . . . . .	9



---

## Práctica 1

# Introducción. Visualización de modelos PLY

### 1.1. Objetivos

Con esta práctica se quiere que el alumno aprenda:

- A utilizar las primitivas de dibujo de OpenGL
- A distinguir la diferencia entre definir un modelo en código y definirlo mediante datos
- A crear estructuras de datos apropiadas para su uso en visualización de modelos gráficos
- A leer modelos guardados en ficheros externos y su visualización, en concreto en formato PLY

### 1.2. Desarrollo

Para el desarrollo de esta práctica se entrega el esqueleto de una aplicación gráfica basada en eventos, mediante glut, y con la parte gráfica realizada por OpenGL. Para facilitar su uso, la aplicación permite abrir una ventana, mostrar unos ejes y mover una cámara básica. Así mismo, se incluye el código básico para dibujar los vértices de un cubo unidad. Se entrega también el código de un lector básicos de ficheros PLY compuestos únicamente por triángulos.

El alumno estudiará el código. En particular debe fijarse en el código que permite dibujar los 8 vértices. Se debe comprobar la rigidez que impone el definir el modelo mediante código (¿qué pasa si en vez de un cubo es un dodecaedro?). Se debe entender que la solución pasa por independizar la visualización de la definición de modelo. Reflexionar sobre esto pues es un concepto muy importante.

El alumno debe encontrar una solución que permita una mayor flexibilidad. Una vez entendido e implementado, se hará uso del código que permite leer un fichero PLY, visua-

lizándolo. Es probable que se vea la necesidad de crear estructuras de datos que faciliten el manejo de los modelos y sus datos. Por ejemplo, en vez de tener 3 flotantes para definir las coordenadas de un vértice, utilizar una estructura que los convierta en una entidad (ver el código `vertex.h` como ejemplo).

Finalmente se creará el código que permita visualizar el modelo con los siguiente modos:

- Alambre
- Sólido
- Ajedrez

Para poder visualizar en modo alambre (también para el modo sólido y ajedrez) lo que se hace es mandar a OpenGL como primitiva los triángulos, `GL_TRIANGLES`, y cambiar la forma en la que se visualiza el mismo mediante la instrucción `glPolygonMode`, permitiendo el dibujar los vértices, las aristas o la parte sólida.

Para el modo ajedrez basta con dibujar en modo sólido pero cambiando alternativamente el color de relleno.

### 1.3. Evaluación

Para evaluación de la práctica se tendrán en cuenta los siguientes items que deberán cumplirse (sobre 10):

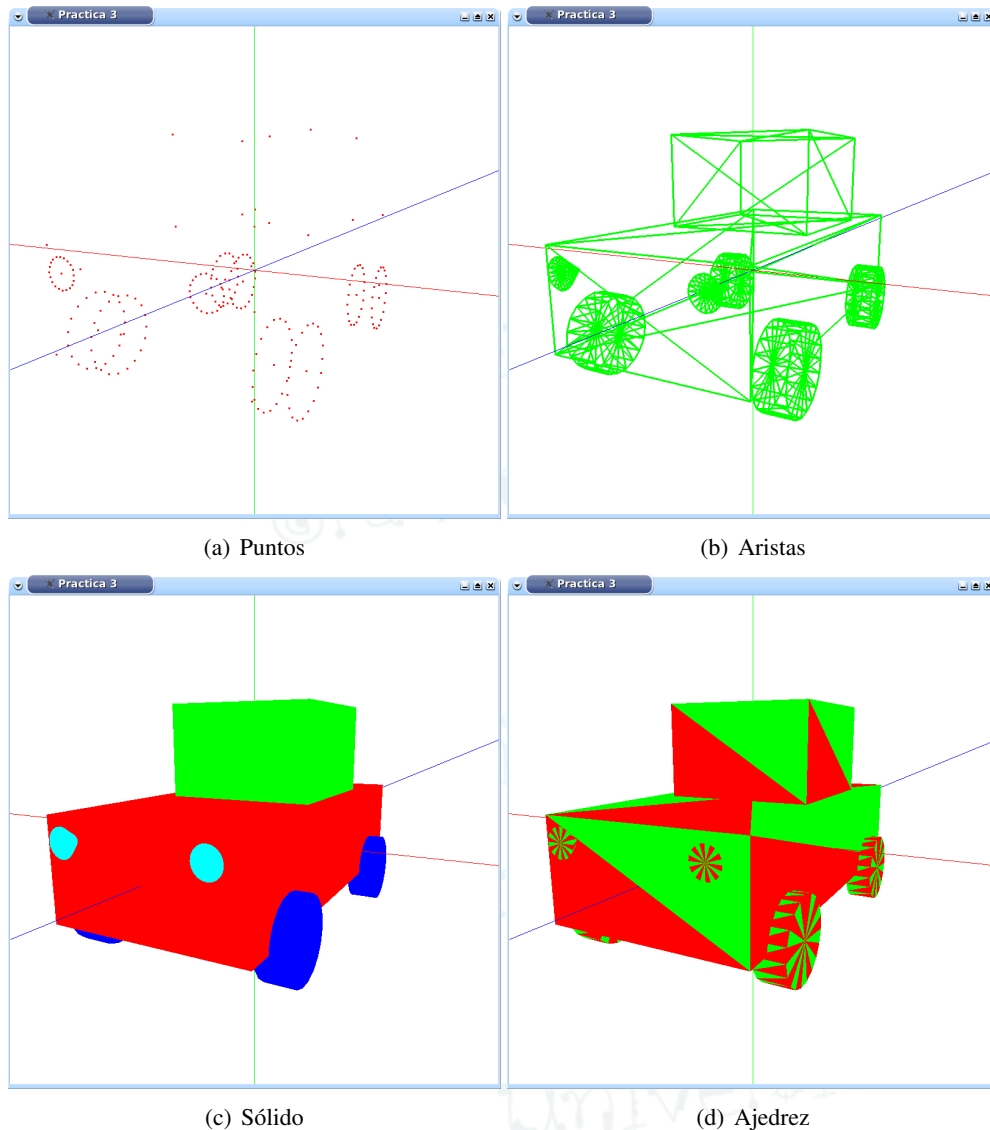
- Creación del código que permite visualizar un vector de flotantes en modo puntos (2pt)
- Lectura de un fichero PLY y visualización en modo puntos (4 pt.)  
Se leerá el fichero PLY y dados los vectores de coordenadas de los vértices y el vector de los índices de vértices, se creará una estructura de datos que guarde los vértices y las caras (se recomienda usar STL y el fichero auxiliar `vertex.h`).
- Creación del código que permite visualizar en los modos alambre, sólido y ajedrez. (4pt)  
Creación de los procedimientos para visualizar en los distintos modos.

### 1.4. Extensiones

### 1.5. Duración

La práctica se desarrollará en 2 sesiones

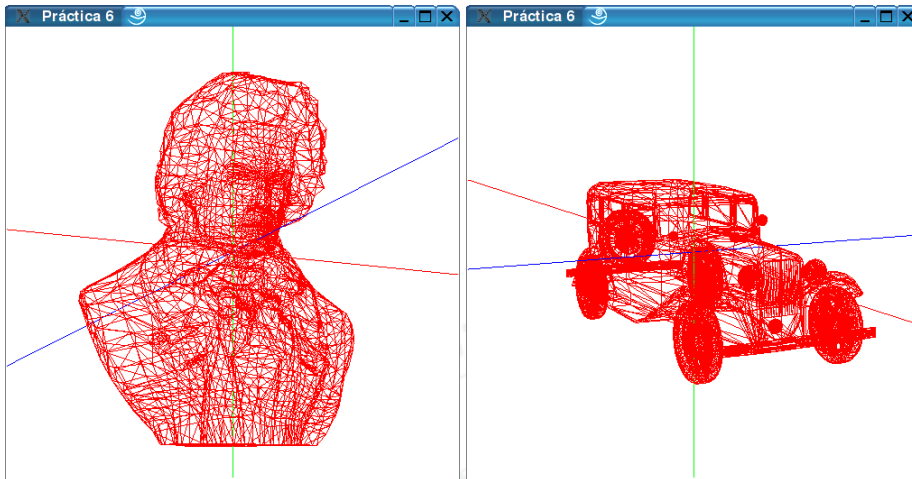




**Figura 1.1:** Coche mostrado con los distintos modos de visualización.

## 1.6. Bibliografía

- Mark Segal y Kurt Akeley; *The OpenGL Graphics System: A Specification (version 4.1)*; <http://www.opengl.org/>
- Edward Angel; *Interactive Computer Graphics. A top-down approach with OpenGL*; Addison-Wesley, 2000
- J. Foley, A. van Dam, S. Feiner y J. F. Hughes; *Computer Graphics: Principles And Practice, 2 Edition*; Addison-Wesley, 1992



**Figura 1.2:** Objetos PLY.

- M. E. Mortenson; *Geometric Modeling*; John Wiley & Sons, 1985