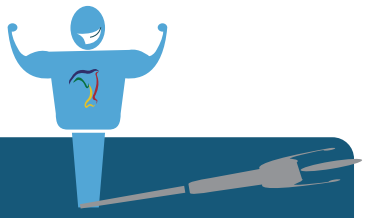
autentia

Avenida de Castilla,1 - Edificio Best Point - Oficina 21B
28830 San Fernando de Henares (Madrid)
tel./fax: +34 91 675 33 06
info@autentia.com - **www.autentia.com**

¿**Qué ofrece** Autentia Real Business Solutions S.L?

Somos su empresa de **Soporte a Desarrollo Informático**.
Ese apoyo que siempre quiso tener...

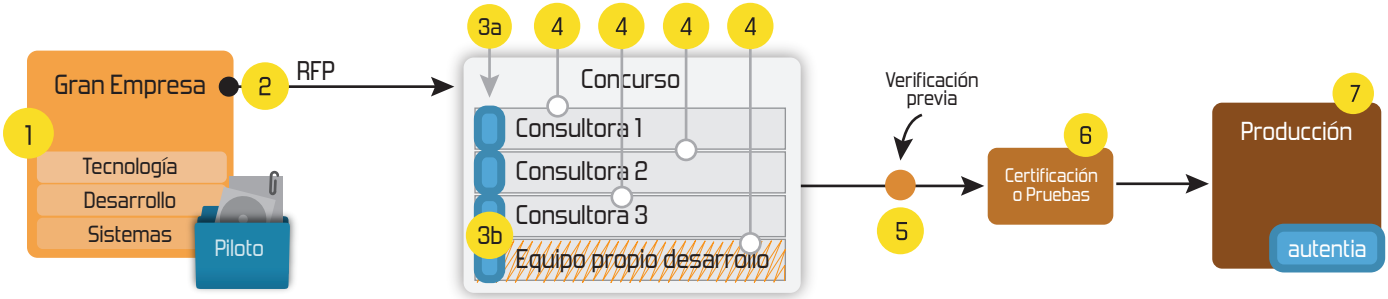
1. Desarrollo de componentes y proyectos a medida



2. Auditoría de código y recomendaciones de mejora

3. Arranque de proyectos basados en nuevas tecnologías

1. Definición de frameworks corporativos.
2. Transferencia de conocimiento de nuevas arquitecturas.
3. Soporte al arranque de proyectos.
4. Auditoría preventiva periódica de calidad.
5. Revisión previa a la certificación de proyectos.
6. Extensión de capacidad de equipos de calidad.
7. Identificación de problemas en producción.



4. Cursos de formación (impartidos por desarrolladores en activo)

Spring MVC, JSF-PrimeFaces /RichFaces, HTML5, CSS3, JavaScript-jQuery

Gestor portales (Liferay)
Gestor de contenidos (Alfresco)
Aplicaciones híbridas

Control de autenticación y acceso (Spring Security)
UDDI
Web Services
Rest Services
Social SSO
SSO (Cas)

JPA-Hibernate, MyBatis
Motor de búsqueda empresarial (Solr)
ETL (Talend)

Dirección de Proyectos Informáticos.
Metodologías ágiles
Patrones de diseño
TDD

Tareas programadas (Quartz)
Gestor documental (Alfresco)
Inversión de control (Spring)

BPM (jBPM o Bonita)
Generación de informes (JasperReport)
ESB (Open ESB)

Compartimos nuestro conocimiento en:
www.adictosaltrabajo.com

Para más información visítenos en:
www.autentia.com

Hosting Patrocinado por enREDados.com

Home | Quienes Somos | Empleo | Foros | Tutoriales | Servicios Gratuitos | Contacte



Tutorial desarrollado por:
Roberto Canales Mora 2003-2005
Creador de AdictosAlTrabajo.com y

Director General de Autentia S.L.

Recuerda que me puedes contratar para echarte una mano:

Desarrollo y arquitectura Java/J2EE
Asesoramiento tecnológico Web
Formación / consultoría integrados en tu proyecto

No te cortes y contacta: 655 99 11 72rcanales@autentia.com.

autentia
real business solutions

Descargar este documento en formato PDF opengl.pdf

Need OpenGL for 3D apps?
Integrate OpenGL in C++ apps with Qt: the cross-platform framework.

Contratar Tarifa Plana
Telefónica y llamadas nacionales ¡Gratis Mp3 o Cámara digital!

Diseño y Desarrollo Web
Desarrollo de sitios optimizados para buscadores, desde €275.00

OpenGL experts
Experienced OpenGL, DirectX, and 3D developers and consultants.

Anuncios Goooooogle Anunciarse en este sitio

Creación de programa OpenGL Mínimo, usando MFC´s

Uno de los mejores modos de construir aplicaciones en tres dimensiones, es la utilización del lenguaje OpenGL (tampoco esta bien dicho....pero bueno...ya que realmente es un intefaz a dispositivos hardware de aceleración de gráficos) .

OpenGL es propiedad de SiliconGraphics y es utilizable en casi todas las plataformas (Unix, NT, etc.), incluyendo consolas.

Hay multitud de recursos para la conversión automática de objetos gráficos tridimensionales (por ejemplo de Autocad) en código C, OpenGL. Así nuestro trabajo consiste en comprender la lógica del lenguaje (para codificar el aplicativo) y dejar a los diseñadores gráficos que hagan su trabajo.

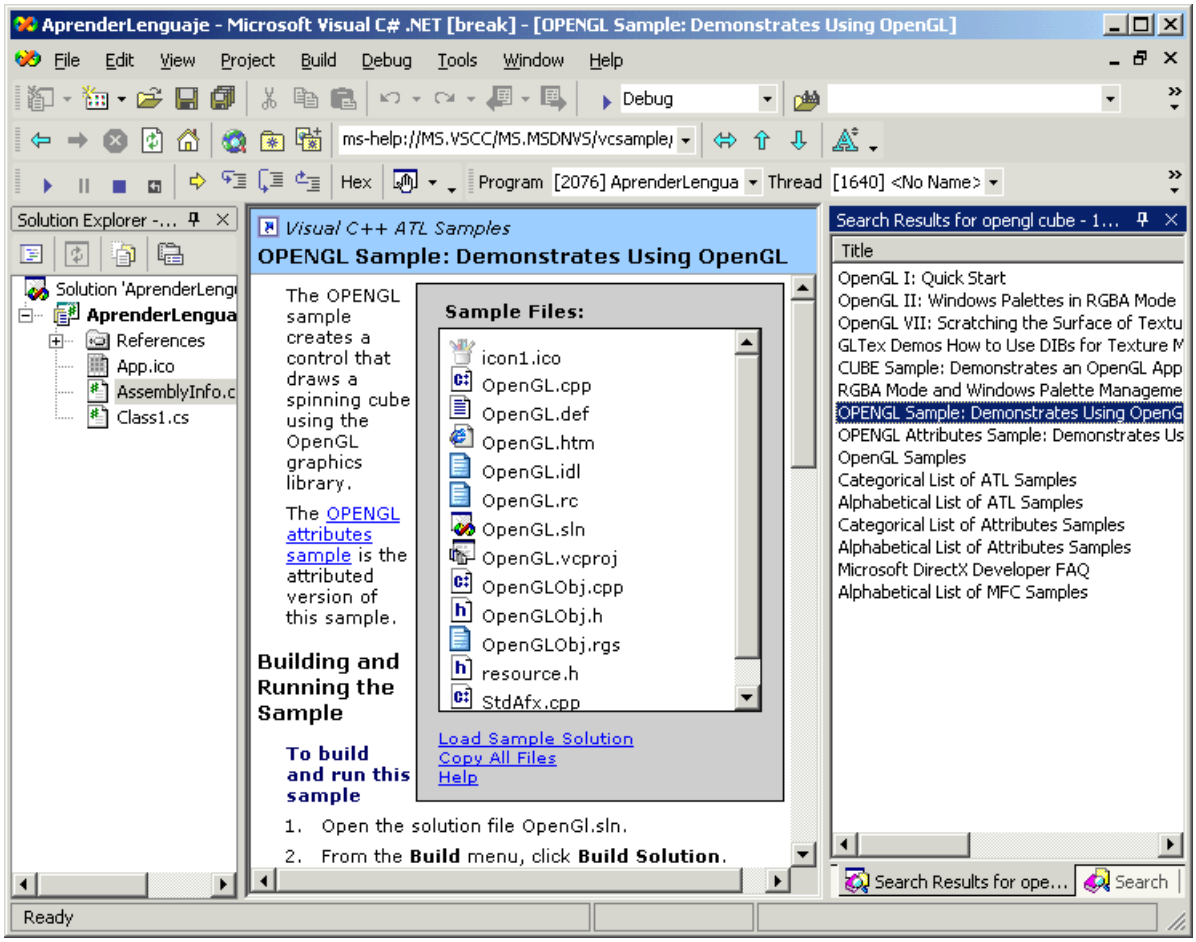
Para comentar la naturaleza del desarrollo OpenGL, vamos a usar el esqueleto de un ejemplo de Visual C++, Cube.....

No es realmente necesario saber Visual C++, sino únicamente restringirnos a meter nuestro código entre dos puntos concretos....al menos para empezar.

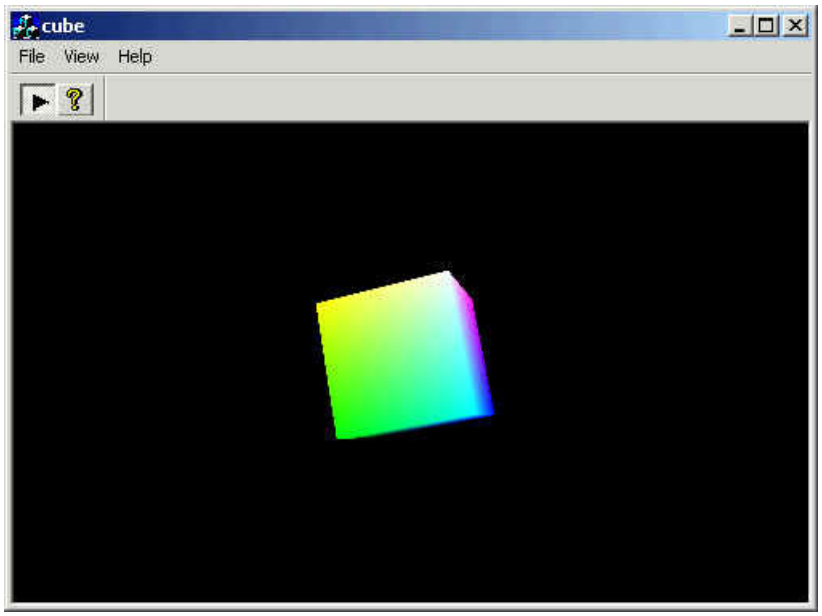
Yo recomiendo que os compresis el libro de OpenGL de Anaya en castellano para empezar ..... aunque ..... la biblia del OpenGL (redbooks) es de Addison Websley (en este link hay una copia digital)

Hay muchos Web dedicados a esto ..... uno de los que más me gusta es NeHe

Más adelante podemos entrar a explicar la naturaleza del programa.



La salida estandar de este programas es:



De todo el código de CUBE, solo nos interesa una porción ... lo demás nos creemos que hace lo que debe hacer para pintar dentro de una ventana ....lo que generemos con nuestro OpenGL

```
void CCubeView::DrawScene(void)

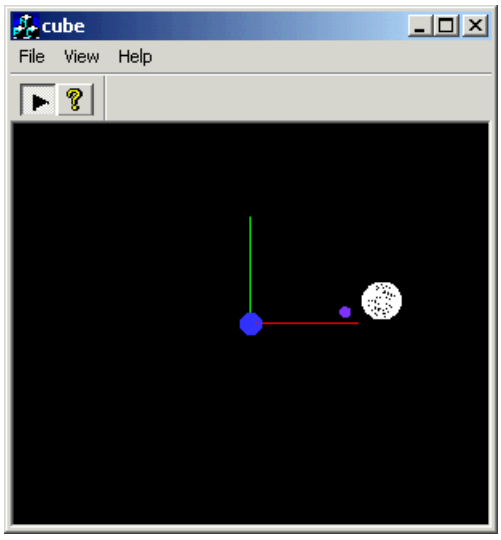
{

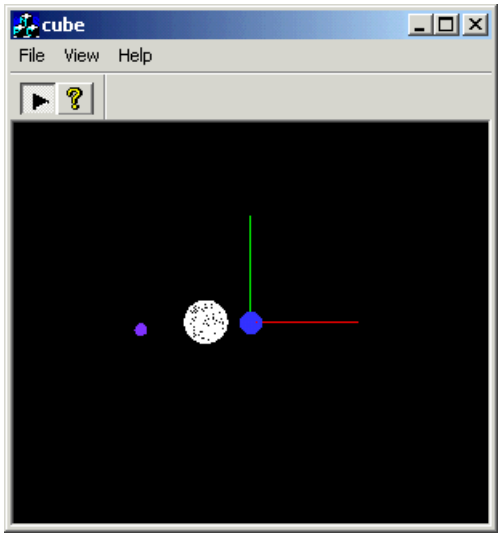
    static BOOL bBusy = FALSE;

    static GLfloat wAngleY = 10.0f;
```

```
    static GLfloat wAngleX = 1.0f;
    static GLfloat wAngleZ = 5.0f;
    if(bBusy)
        return;
    bBusy = TRUE;
    glClearColor(0.0f, 0.0f, 0.0f, 1.0f);
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);
    glPushMatrix();
*****************************************
    Aqui va el código OpenGL
*****************************************
    glPopMatrix();
    glFinish();
    SwapBuffers(wglGetCurrentDC());
    bBusy = FALSE;
}
}
```

Ejemplo





Cuando se ejecuta el botón Play, realmente se esta activando un temporizador que llama periodicamente a nuestra función, la cual va cambiando las variables que nos interesan, y generan una nueva imagen.

El cambio repetitivo de señales, da la sensación de animación (como todas las animaciones)

```
void CCubeView::OnFilePlay()

{

    m_play = m_play ? FALSE : TRUE;

    if (m_play)

        SetTimer(1, 100, NULL);

    else

        KillTimer(1);

}
```

Para tener la capacidad de pintar esferas, se a cogido de otro ejemplo unas lineas de código ... eso permite acceder a librerias con funciones avanzadas OpenGL (hemos ampliado nuestra librería de funciones)

```
#pragma comment(lib, "opengl32.lib")

#pragma comment(lib, "glu32.lib")

#pragma comment(lib, "glaux.lib")

#include "gl/glaux.h"
```

Para generar este ejemplo, similar a un sistema planetario, solo es necesario este código ... parece que todo el mundo pone el mismo ejemplo.....pero la verdad es que se ve perfectamente

```
///////    glRotatef(20, 0.0f, 0.0f, 1.0f); ////// esto para el siguiente ejemplo

glTranslatef(0.0f, 0.0f, -m_fRadius);

glBegin(GL_LINES);    // pintar los ejes X

    glColor3f(1.0f, 0.0f, 0.0f);

    glVertex3f(0.0f, 0.0f, 0.0f);

    glVertex3f(1.0f, 0.0f, 0.0f);
```

```
glEnd();
glBegin(GL_LINES); // pintar los ejes Y
    glColor3f(0.0f, 1.0f, 0.0f);
    glVertex3f(0.0f, 0.0f, 0.0f);
    glVertex3f(0.0f, 1.0f, 0.0f);
glEnd();
glBegin(GL_LINES); // pintar los ejes Z
    glColor3f(0.0f, 0.0f, 1.0f);
    glVertex3f(0.0f, 0.0f, 0.0f);
    glVertex3f(0.0f, 0.0f, 1.0f);
glEnd();

glRotatef(20, 1.0f, 0.0f, 0.0f);

glColor3f(0.2f, 0.2f, 2.0f);
auxWireSphere(0.1); // pintar primera esfera
glRotatef(wAngleY, 0.0f, 1.0f, 0.0f);
glTranslatef(1.0f, 0.0f, 0.0f );

glColor3f(0.5f, 0.2f, 1.0f);
auxWireSphere(0.05);  // pintar segunda esfera

glRotatef(wAngleY, 0.0f, 1.0f, 0.0f);
glTranslatef(0.6f, 0.0f, 0.0f );

glColor3f(1.0f, 1.0f, 1.0f);
auxWireSphere(0.2);  // pintar tercera esfera

wAngleX += 1.0f;    // ALGORITMO DE CAMBIO
wAngleY += 10.0f;
wAngleZ += 5.0f;
```

Descargarse código completo

Para entender los conceptos de OpenGL, probablemente sea necesario que se haya estudiado matemáticas a nivel de 3º Bup o COU (para los que somos mas viejos que ESO ;))

La representación se objetos tridimensionales en superficie plana se basa en la proyección.

Los cálculos se basan en teoría de matrices y ecuaciones trigonométricas....por lo que se oirá hablar mucho de matriz identidad.

Como funciona esto .. a grandes rasgos

Partimos de un sistema de coordenadas x,y,z donde el eje x,y seria el plano de nuestra pantalla.

Tenemos dos comandos básicos

glTranslatef(0.0f, 0.0f, -m_fRadius); // que mueve el origen del sistema de coordenas ...respecto a nosotros y

glRotatef(n, x, y,z);

que rota el sistema en n unidades (radianes, por lo que un angulo de giro va de 0 a 2Pi....y multiplos), usando como vector de giro x,y,z

Es decir, vamos cambiando transladando y rotando el sistema de coordenadas y pintamos siempre respecto al punto 0,0,0..

Otra vez.....

- Imaginar que partimos del punto 0,0,0 donde son x,y,z. El plano x e y son nuestra pantalla.
- El plano Z, coordenadas negativas son en dirección al tubo de reyos catódicos (para los que no tenemos pantalla de plasma)
- Si pintamos una linea de 0,0,0 a 10,10,10 , sera una visectriz a los tres ejes .....pero como la Z es positiva ...no la veriamos
- Si desplazamos el eje de coordenadas un poco hacia la parte negativa de Z (glTranslatef(0.0f, 0.0f, -m_fRadius);) y pintamos lo mismo ....pues ya somos capaz de verlo ....
- Si antes de pintar .... rotamos sobre el eje x (glRotatef(2*Pi/angulo, 1.0f, 0.0f, 0.0f);) realmente estamos metiendonos debajo de la mesa y mirar nuestro eje de coordenadas...desde abajo ....como obviamente...no nos movemos ..... significa que lo hemos inclinado todo lo que se pinte a continuación
- Es decir , nosotros siempre pintamos sobre un eje cartesiano.....y lo que hacemos es transladar o rotar el origen de ese eje.

Bueno.....lo mejor es experimentar ...ya que con 4 lineas te apañas ..... y no cambiar mucho los valores de las variables....hasta comprender el concepto de perspectiva y cubo de proyección (en otros capitulos)

Puede resultar complicado...pero con un ejemplo se ve facil, con una sola linea al principio del bloque de código (activando el código en rojo glRotatef(20, 0.0f, 0.0f, 1.0f);) del sistema de planetas vemos , de la imagen a, el cambio a la imagen b

Hemos girado sobre el eje Z

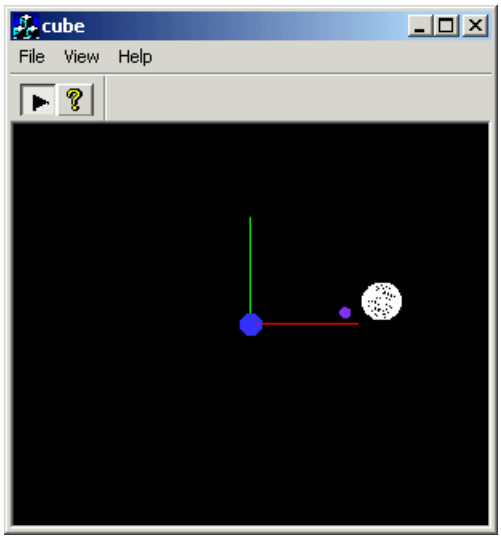


Imagen a) sin cambios

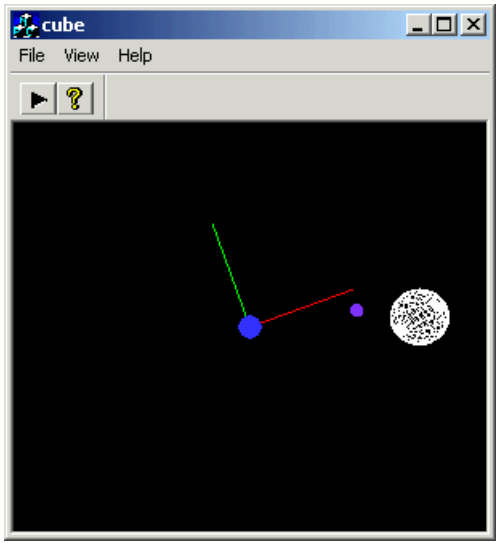


Imagen B) activando..... glRotatef(20, 0.0f, 0.0f, 1.0f);

Sobre el Autor ..

Ultimo cambio 16/12/2002 22:23:14

Si desea contratar formación, consultoria o desarrollo de piezas a medida puede contactar con

Creatividad Internet

Autentia S.L. Somos expertos en:
J2EE, C++, OOP, UML, Vignette, Creatividad ..
y muchas otras cosas

Nuevo servicio de notificaciones

Si deseas que te enviemos un correo electrónico cuando introduzcamos nuevos tutoriales, inserta tu dirección de correo en el siguiente formulario.

Subscribirse a Novedades	
e-mail	
	Enviar

Otros Tutoriales Recomendados (También ver todos)

Nombre Corto	Descripción

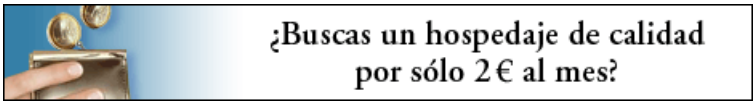
Nota: Los tutoriales mostrados en este Web tienen como objetivo la difusión del conocimiento.

Los contenidos y comentarios de los tutoriales son responsabilidad de sus respectivos autores.

En algún caso se puede hacer referencia a marcas o nombres cuya propiedad y derechos es de sus respectivos dueños. Si algún afectado desea que incorporemos alguna reseña específica, no tiene más que solicitarlo.

Si alguien encuentra algún problema con la información publicada en este Web, rogamos que informe al administrador rcanales@adictosaltrabajo.com para su resolución.

Patrocinados por enredados.com .... Hosting en Castellano con soporte Java/J2EE



www.AdictosAlTrabajo.com Opimizado 800X600