

Utiliser OpenGL et GLUT avec Dev-C++

Il y a quelque temps cette page n'était qu'une réponse – assez hésitante – à une question d'un étudiant. Depuis elle me vaut un courrier régulier, moitié « merci pour le tuyau ! », moitié « pourquoi ça ne marche pas chez moi ? » qui m'oblige à vérifier ce que j'y raconte. Je m'en passerais bien – OpenGL ne faisant pas partie de mes occupations habituelles – mais, comme la documentation sur l'installation de GLUT n'est pas très copieuse, j'essaie de garder cette note à peu près juste. Quoi qu'il en soit :

- les informations données ici le sont sans *aucune garantie*,
- je suis preneur de *tout avis autorisé* corrigeant mes erreurs ou comblant mes lacunes.

1 Si vous développez avec Visual C++ de Microsoft alors vous avez d'office OpenGL, qui est inclus dans les bibliothèques de base. Je pense que la situation est la même si vous utilisez les compilateurs C++ de Borland, mais je n'ai pas essayé.

Il vous faut aussi GLUT (OpenGL Utility Toolkit). Le plus simple est d'en télécharger la version courante depuis [le site de Nate Robins](#). Cela vous procurera trois fichiers indispensables : `glut.h`, `glut.lib` et `glut.dll`. Copiez-les dans votre dossier de travail, puis incluez les deux premiers dans votre projet (*Add To Project > Files...*). J'ai essayé avec Visual C++ 6, cela marche très bien.

De plus, Nate Robins fournit tous les sources de GLUT, avec des fichiers *project* et des fichiers *makefile* parfaitement opérationnels sous Visual C++.

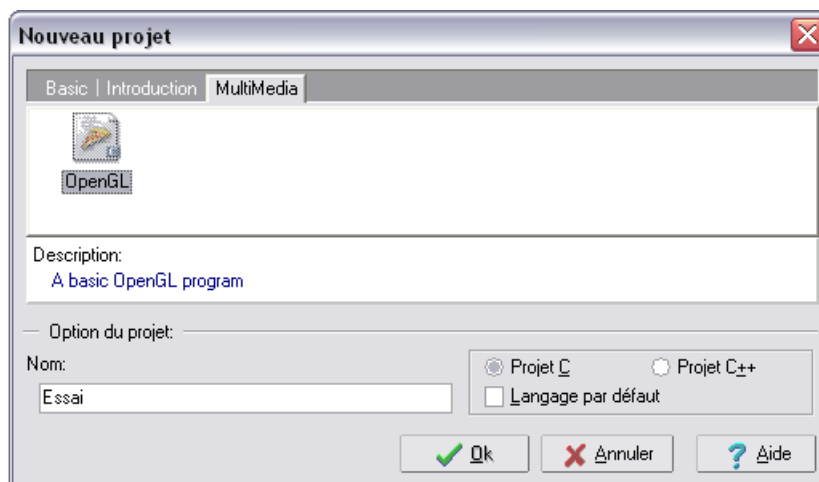
2 Parlons désormais du développement avec Dev-C++. Je suppose que vous avez la dernière version de ce produit qui, en novembre 2006, est la version 4.9.9.2. Cela se trouve sur [le site de Bloodshed](#), s'appelle « Dev-C++ 5.0 beta 9.2 (4.9.9.2) with Mingw/GCC 3.4.2 » et pèse 9 Mo environ. Le cas échéant, lisez les [instructions pour installer Dev-C++](#).

3 Si vous ne l'avez pas encore fait, apprenez à vous servir de Dev-C++ en employant un *projet* (ce qui, OpenGL ou pas, est *la bonne manière* d'utiliser Dev-C++). Mode opératoire : lancez Dev-C++ puis créez un projet :

- commande *Fichier > Nouveau > Projet...*
- dans la boîte de dialogue *Nouveau projet*, choisissez *Projet C* (si vous programmez en C) et *Basic*, *Console Application* (ne choisissez pas *Windows Application*, si vous saviez écrire des applications Windows vous ne seriez pas en train de lire cette page !),
- cochez la case *Langage par défaut* à moins que vous ayez une raison de ne pas le faire,
- dans la case *Nom* saisissez celui que vous voulez,
- après avoir cliqué sur *Ok* on vous demandera d'enregistrer le projet : sauvez-le dans le dossier où vous envisagez de travailler.

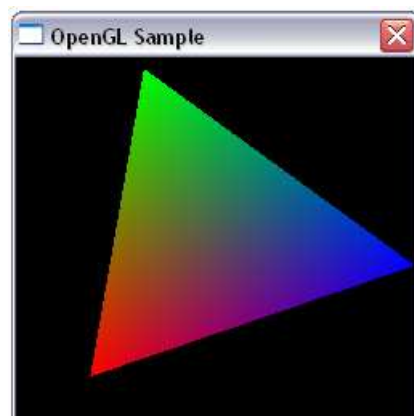
A partir de ce moment, vous possédez un projet, ce qui vous permet de compiler des applications ayant plusieurs fichiers sources et/ou des bibliothèques particulières. Pour ajouter des fichiers sources au projet : *Projet > Nouveau fichier source* (si le fichier est nouveau) ou bien *Projet > Ajouter au projet* (si le fichier existe déjà).

4 Sans rien faire de plus, vous disposez déjà de la bibliothèque OpenGL. Elle se trouve dans le dossier **Dev-C++/lib** et s'appelle **libopengl32.a**. Pour vous en convaincre, créez un nouveau projet en spécifiant *Multimedia*, *OpenGL* au lieu de *Basic*, *Console Application* :



Compilez et exécutez le petit programme de démonstration que Dev-C++ a mis à titre d'exemple dans le nouveau projet :





De plus, ce triangle tourne ! Intéressant, n'est-ce pas ?

5 Germain Barret (grand merci !) signale un procédé pour installer GLUT nettement plus simple et fiable que le bricolage que je décris dans la section 5bis ci-dessous. Cela consiste à utiliser un *DevPak*, c'est-à-dire une archive ayant un certain format reconnu par Dev-C++, lequel se charge d'installer les fichiers des bibliothèques dans les bons répertoires.

De plus, la commande correcte pour l'éditeurs de liens est automatiquement mise en place lorsqu'on crée un projet (commande *Fichier > Nouveau > Projet... > MultiMedia*) de type « glut ».

On trouve des *devpak* à jour concernant GLUT – ainsi que GLUI, OpenGLUT, etc. – sur le site <http://www.nigels.com/glt/devpak/>. Je vous conseille de télécharger le fichier qui, en novembre 2006, s'appelle « glut.3.7.6+.DevPak » et pèse une centaine de Ko.

Une fois le *devpak* téléchargé, pour l'installer il faut rentrer dans Dev-C++ et utiliser la commande *Outils > Package Manager > Install*.

N.B. 1 – Avec ce *devpak* vous pouvez créer des projets de type « glut » comme indiqué (commande *Fichier > Nouveau > Projet... > MultiMedia > glut*) mais ce n'est pas une obligation. Vous pouvez aussi bien utiliser des projets « Console » (commande *Fichier > Nouveau > Projet... > Basic > Console Application*) ou même des projets vides (commande *Fichier > Nouveau > Projet... > Basic > Empty Project*) à la condition de suivre alors les indications de la section 6 ci-dessous à propos des options pour l'édition de liens.

N.B. 2 – Nigel Stewart, le fournisseur du *devpak* dont nous parlons, signale qu'il est nécessaire de définir **GLUT_STATIC** comme un *symbole du préprocesseur*. J'ai constaté cela ne nuit pas, mais je n'ai pas d'exemple montrant que c'est indispensable. Si vous souhaitez faire une telle définition, allez dans *Projet > Options du projet > Paramètres* et écrivez le texte « **-DGLUT_STATIC** » dans les deux panneaux *Compilateur* et *Compilateur C++*.

5bis Si vous n'avez pas pu suivre les indications du n° 5, il vous faut installer la bibliothèque GLUT (et aussi GLU, qui porte bien son nom !). Téléchargez les fichiers suivants (même si vous les avez déjà, car plusieurs versions circulent, plus ou moins cohérentes entre elles) :

[glut.h](#)
[libglu32.a](#) et [libglut32.a](#)
[glut32.dll](#)
[tp1exo1.c](#), [tp1exo2.c](#),
[bigtest.c](#) et [tests.zip](#)

[Tous ces fichiers en une seule archive zip](#)

Copiez (en écrasant peut-être des fichiers de même nom qui se trouvent dans ces dossiers) :

`glut.h` dans le dossier `Dev-Cpp/include/GL`
`libglu32.a` et `libglut32.a` dans le dossier `Dev-Cpp/lib`
`glut32.dll` dans le dossier `Dev-Cpp/bin`

Attention : **Dev-Cpp** est le dossier d'installation de Dev-C++ ; chez vous il porte peut-être un autre nom.

6 Dans vos programmes, écrivez la directive pour inclure `glut.h` sous la forme

```
#include <GL/glut.h>
```

D'autre part, faites la commande *Projet > Options du projet*. Dans la boîte de dialogue *Options du projet* choisissez l'onglet *Paramèt...* Dans la zone de texte *Editeur de liens* écrivez (sans vous faire aider par le bouton *Ajouter fichier*) :

```
-lglut32  
-lopengl32  
-lglu32  
-lglut32  
-lwinmm
```

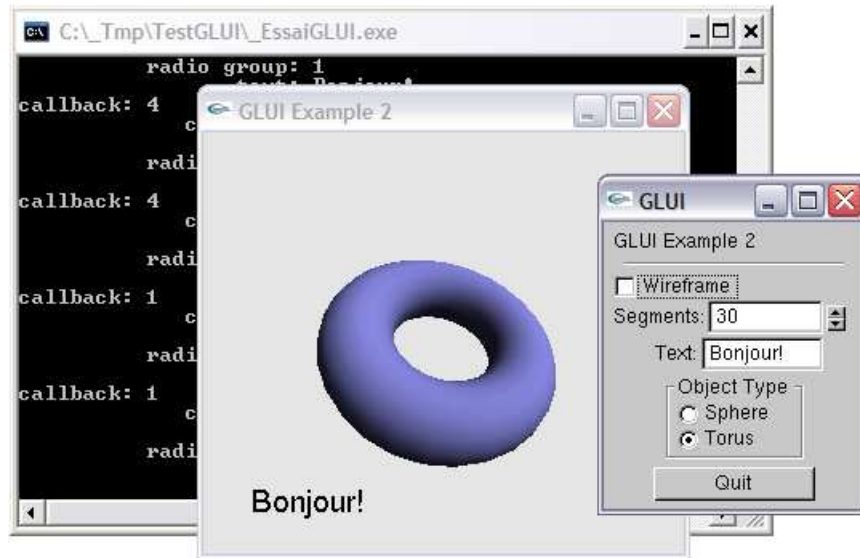
(liste récemment complétée, merci à Emmanuel Romagnoli) Maintenant, vos programmes doivent se compiler correctement. De plus, si vous

(liste récemment complétée, merci à Emmanuel Romagnoli) Maintenant, vos programmes doivent se compiler correctement. De plus, si vous avez copié le fichier `glut32.dll` dans un des dossiers indiqués par la variable d'environnement `path` (par exemple `Dev-Cpp/bin` comme suggéré précédemment) votre programme doit s'exécuter correctement.

Si ce n'est pas le cas, c'est probablement que `Dev-Cpp/bin` ne figure pas parmi les dossiers indiqués par la variable `path`. Modifiez la valeur de cette variable ou, plus simplement, copiez le fichier `glut32.dll` dans le dossier contenant votre fichier exécutable. Vous pouvez aussi copier `glut32.dll` dans un des « dossiers à dlls » de Windows, comme `C:\WINDOWS` ou `C:\WINDOWS\system`.

Pour vous assurer que votre installation fonctionne, compilez et exécutez les exemples `tp1ex1.c` et `tp1ex2.c`. Si tout va bien, compilez et exécutez l'exemple `bigtest.c` : vous serez fiers de vous !

7 Compléments : GLUI. GLUI est une bibliothèque portable, car entièrement construite au-dessus de GLUT, qui facilite significativement la construction d'interfaces-utilisateur graphiques :



Voici comment installer et utiliser GLUI avec Dev-C++ :

1. Téléchargez GLUI à l'adresse <http://sourceforge.net/projects/glui/>. Vous obtenez une archive zip contenant les sources et le manuel (format PDF) de GLUI
2. Décompressez cette archive. En plus des sources et du manuel, elle contient un fichier *makefile* permettant d'utiliser GLUI sous UNIX/Linux et un dossier *msvc* permettant de l'utiliser sous Windows avec Visual C++
3. Pour l'utiliser avec Dev-C++ assurez-vous que vous disposez de la dernière version et que vous savez le configurer pour l'utilisation de GLUT (points 2 à 6, ci-dessus)
4. Créez un nouveau projet Dev-C++ : dans la boîte *Nouveau projet* choisissez *Basic*, *Console Application* et *Projet C++*
5. Ne pas oublier d'aller dans *Projet > Options du projet*, onglet *Paramètres*, zone *Editeur de liens* et y inscrire, comme chaque fois qu'on utilise GLUT : `-lopengl32 -lglu32 -lglut32`
6. Ajoutez au projet (menu *Projet*, commande *Ajouter au projet*) un des fichiers `exemple<i>.cpp`, tous les autres fichiers `.cpp` de l'archive, aucun fichier `.c` et aucun fichier `.h` (mais ces fichiers `.c` et `.h` sont utilisés et doivent se trouver dans le dossier contenant les `.cpp`)
7. Dans chacun des cinq fichiers `exemple<i>.cpp` il faut
 - remplacer « `void main` » par « `int main` »,
 - ajouter « `return 0;` » juste avant l'accolade finale de la fonction `main`.

Avec cette seule modification (d'autres compilateurs sont plus tolérants concernant le type de la fonction `main`) les cinq exemples `exemple<i>.cpp` se compilent sans le moindre warning et s'exécutent. Et ils sont assez spectaculaires!

Attention. Fermer la fenêtre ne « tue » pas l'application. Cela est un défaut connu de GLUT pas encore corrigé dans la version que vous téléchargez ici.