Qt OpenGL - Gestion des extensions avec QGLContext::getProcAddress()

Sommaire

Gérer les extensions

Les fonctionnalités offertes par les cartes graphiques sont très variables en fonction du constructeur et évoluent à un rythme différent d'OpenGL. Pour gérer cette grande diversité et pour faciliter l'intégration de nouvelles fonctionnalités, OpenGL utilise un système d'extensions : chaque nouvelle fonction est ajoutée dans une extension. Pour pouvoir l'utiliser, il faut donc : vérifier que le matériel supporte la fonctionnalité puis charger la fonction. Le lecteur se reportera au tutoriel Les extensions OpenGL pour avoir des explications détaillées sur la procédure générale à suivre, en particulier sur la syntaxe à utiliser pour déclarer les pointeurs de fonctions.

Il est possible d'utiliser une bibliothèque tierce qui permet de prendre en charge les extensions OpenGL, par exemple GLEW. Dans le cas où l'on

souhaite utiliser peu d'extensions et que l'ajout d'une bibliothèque est trop lourd, Qt fournit la fonction getProcAddress pour récupérer une extension OpenGL :

```
    PFNGLGENBUFFERSARBPROC glGenBuffers = (
PFNGLGENBUFFERSARBPROC) getProcAddress("glGenBuffersARB");
    if (glGenBuffers)
        // la fonction est disponible
    else
        // la fonction n'est pas disponible
```

Creer une classe QOpenGLFunctions

Fonctions OpenGL disponibles dans les classes derivees de QOpenGLFunctions. Par exemple pour les fonctions du core OpenGL 3.3 : http://doc-snapshot.qt-project.org/qt5-dev/qopenglfunctions-3-3-core.html

Si vous voulez utiliser des fonctions de GL supérieur, vous pouvez utiliser les extensions ou créer votre propre classe QOpenGLFunctions. On va voir comment faire, pour OpenGL 4.5 (la dernière version actuelle)>

En interne de Qt, ces classes sont générées a partir d'un script, qui prend un fichier XML publie par Khronos et qui contient la liste des API pour chaque version de Qt.

- Utiliser ce script ?
- créer une nouvelle classe a partir de QOpenGLFunctions ?
- créer une nouvelle classe a partir de QOpenGLFunctions_4_3_Core ?

OpenGL, Qt