Subject: [GLCtrl] Adding an Initilisation function to GLPaint
Posted by Xemuth on Tue, 03 Mar 2020 10:32:17 GMT
View Forum Message <> Reply to Message

Hello community,

When we have to deal with OpenGL, it is often interesting to load a certain amount of data (Texture / Vertex / Shaders) before rendering a scene.

Nowaday, to load thoses things, I must declare a boolean in my derived class of GLCtrl and use it to perform many load at the first loop in GLPaint, as Follow :

```
 bool isLoaded = false;
 virtual void GLPaint() {
  if(!isLoaded){
   /* Loading Few things  ! */
   isLoaded=true;
  }

  /* Code to perform draw */
  Refresh();
 }
```

Even if this way of working is functional, Having to write many and many code and routine of loading in GLPaint() is annoying and don't help in code readability.

That's why I started to dig out the GLCtrl to implement a virtual function called Initialisation() wich can be overwritted. The purpose of it is to be called right after OpenGL context Initialisation. My first way of doing this was Adding the call of function here (GLCtrl.cpp line 33):

```
void GLCtrl::Init()
{
 Transparent();
#ifdef PLATFORM_WIN32
 pane.ctrl = this;
 Add(pane.SizePos());
#endif
 restore_gl_viewport__ = SetCurrentViewport;
 Initialisation(); //--> I call Initialisation Here
}
```

But (It seems) due to Ctrl creation, the function wich is called at this moment is the non overwritted Initialisation() from GLCtrl and not the one I have overwritted in my derived class from GLCtrl.

So my next try was to implement it here (Win32GLCtrl.cpp) :void GLCtrl::GLPane::State(int reason)

```
{
 DHCtrl::State(reason);

 if(reason == OPEN) {
  HWND hwnd = GetHWND();
  CreateContext();
  HDC hDC = GetDC(hwnd);
  if(!SetPixelFormat(hDC, s_pixelFormatID, &s_pfd))
   return;
  ReleaseDC(hwnd, hDC);
  static_cast<GLCtrl*>(parent)->Initialisation();//--> I call Initialisation Here
 }
}
```
This time, it call the overwritted Initialisation() but OpenGL Context is not yet well initialised, consequence is every call to OpenGL api (all call to allocate some memory in vram) fail !

So far I dig, I didn't find the proper way of adding this function, so I need your help ! If someone have an idea...

Thanks in advance.
Best regard

Xemuth