

# INF2705 Infographie

## Tutoriel

### *Débuggage avec RenderDoc*

# 1 Débuggage avec RenderDoc

RenderDoc est un outil de debug open source pour les applications graphiques. Il est compatibles avec plusieurs API. Pour OpenGL, il faut avoir un contexte en mode core plus grand que 3.3 (ce qu'on utilise en ce moment). Il y a d'autres alternatives, comme Nvidia Nsight qui permet aussi de faire du débbugage, mais principalement sur les cartes nvidia.

Bref, si vous voulez l'utiliser, vous devez partir l'application directement dans RenderDoc. Mettre les path de l'executable et le working directory pour qu'il puisse charger les fichiers correctement (dans notre cas le dossier '/src/'). Une fois dans l'application, on peut cliquer sur F12 pour faire une capture de la frame.

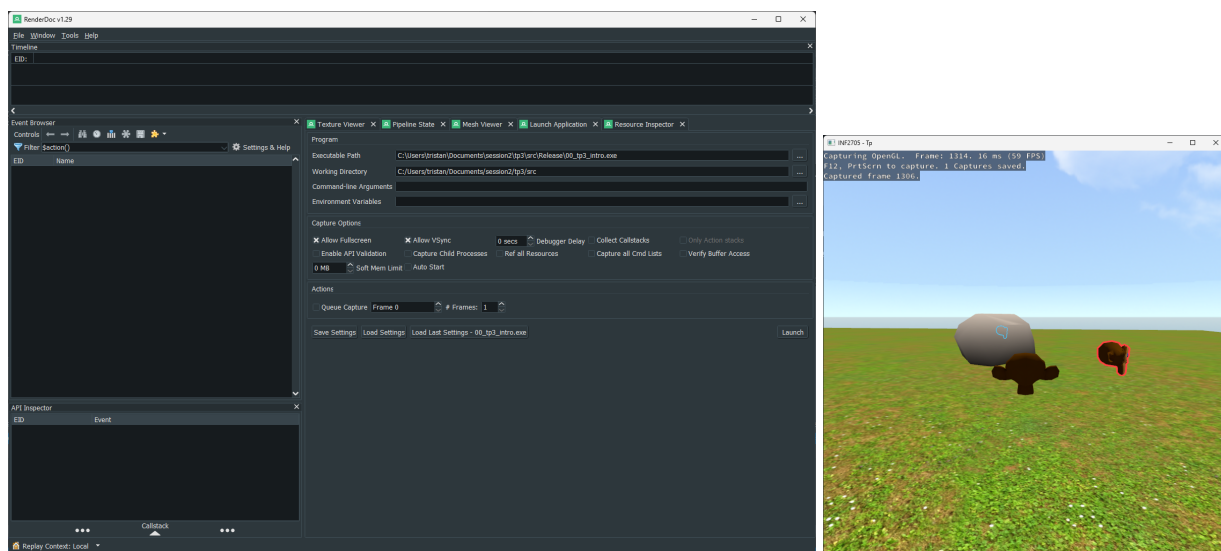


FIGURE 1 – Ouverture de l'application à profiler.

Un nouvel onglet devrait s'être ouvert avec les captures de l'application. Si double clic dessus, on a accès la liste des évènements graphiques. On peut les sélectionner et visualiser l'effet qu'ils ont eus sur les buffers dans l'onglet Texture Viewer.

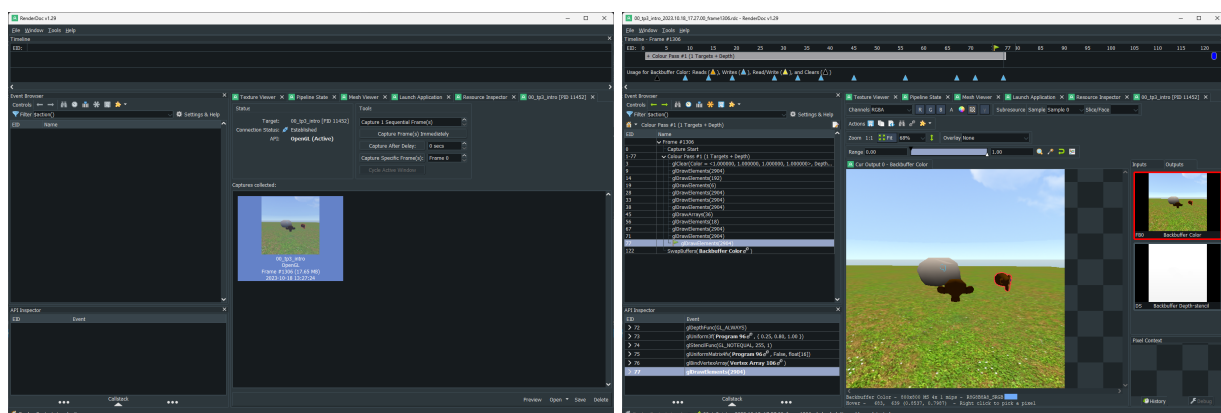


FIGURE 2 – Chargement de la capture et analyse du frame buffer.

Pour débbugger le stencil, on peut le sélectionner à droite, puis cocher Stencil en haut. Les couleurs seront mises en mode automatique. Avec l'icône de baguette, on peut ajuster le range automatique pour voir ce qui se passe sur le stencil. En survolant avec le curseur de la souris ou clic droit, on peut aussi voir la valeur directement dans le buffer en hexadécimal (en bas).

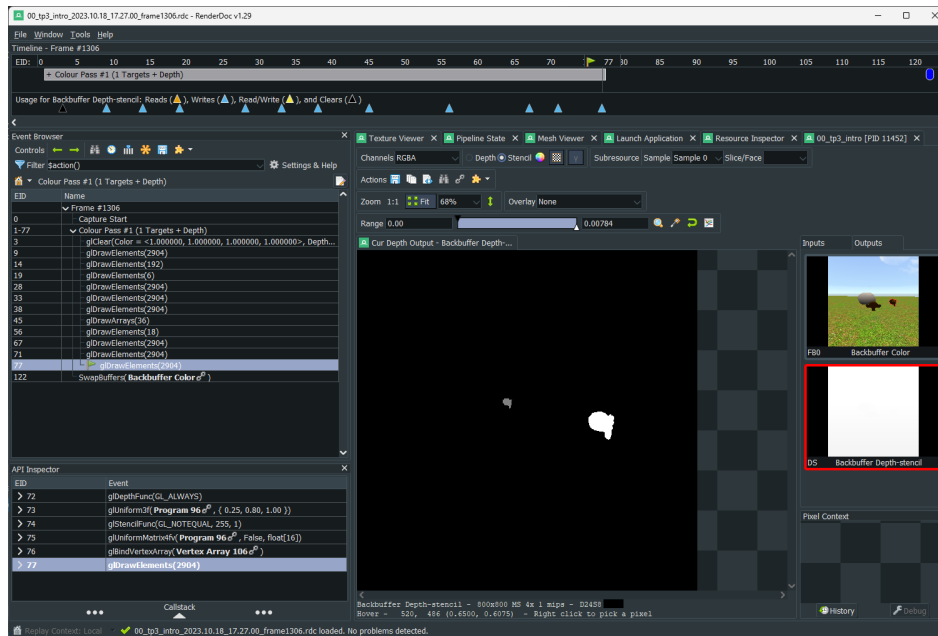


FIGURE 3 – Analyse du stencil buffer.

Tutoriel assez bref, n'hésiter pas à poser des questions sur l'outil. Il permet de faire bien plus que de débbugger le stencil, quoique cela peut s'avérer difficile pour quelqu'un qui débute. Je vous invite à l'explorer.