Display Lists

Noah Geller

Mike Bailey

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License.

You don't want to execute all that code every time you want to redraw the scene, so draw it once, store the numbers in GPU memory, and call them back up later.

The solution is to incur the sphere-creation overhead once, and whenever the sphere needs to be redrawn, just draw the saved numbers, not the equations. This is a Display List.

Creating the Display List in InitLists():

Calling up the Display List in Display():

Without Display List:
The CPU re-computes and transmits the sphere coordinates every time they need to be drawn.

With Display List:
The CPU computes and transmits the sphere coordinates once and then they are grabbed from GPU memory every time they need to be drawn.

You don't want to execute all that code every time you want to redraw the scene, so draw it once, store the numbers in GPU memory, and call them back up later.

1. How many unique, unused, consecutive DL identifiers to give back to you
2. The ID of the first DL in the unique, unused set
3. The ID of the first DL in the unique, unused set
4. Open up a display list in (GPU) memory
5. All done with storing the numbers in the DL
6. Pull all the coordinates, etc. from memory, just as if the code to generate them had been executed here.

Even worse, the trig calls are inside single or nested for-loops!