OpenGL 3D Graphics in Liberty BASIC

Lesson Four: Display Lists

by Robert McAllister

A disply list is basically an array that holds a list of calls to OpenGL. Since these lists reside in memory they can be executed much faster than making each call one at a time. They are best used for parts of a scene that will not change from frame to frame. You can not modify them but they can be deleted and re-created if needed.

You start with a call to "glGenLists numlists" where numlists is the number of lists you need to set aside memory for. Below, "CALL glNewList 1 , 4865" tells OpenGL that we are starting to add items to display list 1. Now you create the objects you want contained in the list and follow it with a call to "glEndList".

The program below creates a single display list and then displays it on the screen.

```
' Create a display list
CALL
 ClearView eyeX , eyeY , eyeZ , centerX , centerY , centerZ , upX , up
Y , upZ
CALL glGenLists 1 ' reserve memory for 1 display list
CALL glNewList 1 , 4865
CALL glBegin GL.TRIANGLES
  CALL glColor4fv 1 , 0 , 0 , 1
    CALL glVertex -1 , -1 , 0
  CALL glColor4fv 0 , 1 , 0 , 1
    CALL glVertex 0 , 1 , 0
  CALL glColor4fv 0 , 0 , 1 , 1
    CALL glVertex 1 , -1 , 0
CALL glEnd
CALL glEndList
CALL glCallList 1 ' execute the commands in display list 1
CALL RefreshView
WAIT
```

To delete the contents of a display list make a call to "glDeleteLists start , range" where "start" is the number of the first list to delete and "range" is the number of lists to delete.

In this sample, three display lists are created and displayed. After a two second pause, display list 2 is deleted and the scene is redrawn.

```
' create three display lists, a line, a triangle, a square and then de
```

```
lete the triangle
CALL
 ClearView eyeX , eyeY , eyeZ , centerX , centerY , centerZ , upX , up
Y , upZ
CALL glGenLists 3 ' reserve memory for 3 display lists

' create the first display list
CALL glNewList 1 , 4865
CALL glBegin GL.LINES
  CALL glColor4fv 1 , 0 , 0 , 1
    CALL glVertex -1 , -1 , 0
  CALL glColor4fv 0 , 0 , 1 , 1
    CALL glVertex 1 , -1 , 0
CALL glEnd
CALL glEndList

' create the second display list
CALL glNewList 2 , 4865
CALL glBegin GL.TRIANGLES
  CALL glColor4fv 1 , 0 , 0 , 1
    CALL glVertex -1 , -.5 , 0
  CALL glColor4fv 0 , 1 , 0 , 1
    CALL glVertex 0 , 1 , 0
  CALL glColor4fv 0 , 0 , 1 , 1
    CALL glVertex 1 , -.5 , 0
CALL glEnd
CALL glEndList

' create the third display list
CALL glNewList 3 , 4865
CALL glBegin GL.QUADS
  CALL glColor4fv 1 , 0 , 0 , 1
    CALL glVertex -1 , 0 , 0
  CALL glColor4fv 0 , 1 , 0 , 1
    CALL glVertex -1 , 2 , 0
  CALL glColor4fv 0 , 1 , 1 , 1
    CALL glVertex 1 , 2 , 0
  CALL glColor4fv 0 , 0 , 0 , 1
    CALL glVertex 1 , 0 , 0
CALL glEnd
CALL glEndList

'draw the display lists
CALL glCallList 1
CALL glCallList 2
CALL glCallList 3
CALL RefreshView

CALL Pause 2000
' Pause for two seconds so we can see what is happening

CALL glDeleteLists 2 , 1 ' Remove the contents of display list 2

CALL
 ClearView eyeX , eyeY , eyeZ , centerX , centerY , centerZ , upX , up
Y , upZ

'draw the display lists
CALL glCallList 1
CALL glCallList 2 ' display list 2 is now empty
CALL glCallList 3

CALL RefreshView

WAIT
```

For those who have downloaded and run the Star Trek intro in the forum's graphics section, the display list that contains the Enterprise has 1197 triangles which require 6 OpenGL calls each and they are displayed 2031 times. That comes to 14,586,642 calls being executed during the animation. It's a good example of how display lists can speed things up.

In the next lesson we will be looking into "Complex surfaces"