Pyramid

Exercise for learning to program scripts for PADrend

Benjamin Eikel   Claudius Jähn
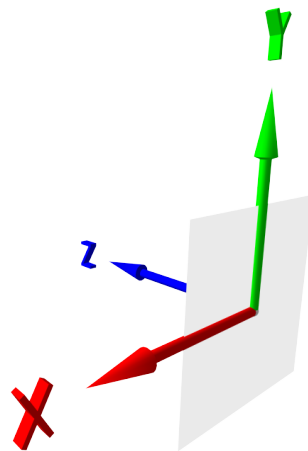
July 25, 2012

Introduction

Figure 1: Rectangle with coordinate system

Given is the `Mesh` of a rectangle, which has a size of one times one units.

A mesh is an array of vertices, an array of indices, and the description of the attributes of the vertices. A vertex is defined by its index inside the array and stores the values of its attributes. Commonly used attributes are a three-dimensional position vector, a three-dimensional normal vector, an RGB color value, and a two-dimensional texture coordinate. The second array of the mesh stores the indices that compose a primitive. For example, if the mesh is used to store triangles, every three indices compose a triangle. In MinSG, to store a mesh inside the scene graph, the mesh is attached to a `GeometryNode`.

Figure 1 shows the rectangle together with a coordinate system. The coordinate system visualizes the placement of the rectangle inside the world coordinate system. The rectanlge lies inside the x-y plane with its center being the origin.

The following script creates the rectangle and attaches it to a node.

```
var rectangleMesh =
        Rendering.MeshBuilder.createRectangle(1.0, 1.0);
var geoNode = new MinSG.GeometryNode();
geoNode.setMesh(rectangleMesh);
```

To create a new root node for a scene, an instance of a `ListNode` is created:

```
var sceneNode = new MinSG.ListNode();
```

A `ListNode` groups multiple nodes – its children – together. To tell PADrend to use a node as the current scene, the node has to be registered as a scene and selected:

```
PADrend.registerScene(sceneNode);
PADrend.selectScene(sceneNode);
```
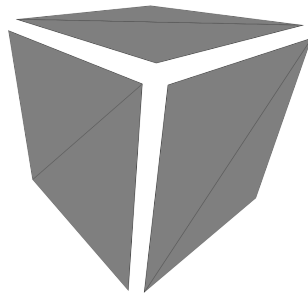
Exercise 1

Figure 2: Exploded view of a cube, which is composed of multiple rectangles

Use the given rectangle six times to create a cube. Rotate and translate the rectangles into the correct position. The cube should look like the one in Figure 2. The following script shows an example on how to create and transform a single side.

```
// Create a new node holding the sides of the cube
var cube = new MinSG.ListNode();
// Create a clone of the node having the rectangle
var leftSide = geoNode.clone();
// Move the node to the left
leftSide.moveLocal(-0.5, 0.0, 0.0);
// Rotate the node by 90 degrees around the y axis
leftSide.rotateLocal_deg(90.0, 0.0, 1.0, 0.0);
// Add the node to the cube node
cube.addChild(leftSide);
// Add the cube to the scene
sceneNode.addChild(cube);
```

Exercise 2

Build a pyramid of cubes. You can reuse the cube node from the previous exercise. Use a scene graph structure, such that every level of the pyramid is a group. An example of a pyramid can be seen in Figure 3.

Exercise 3

Color the pyramid from the previous exercise with a gradient. The top level should be colored red. The bottom level should be colored blue. For each level, only a single material should be used. In MinSG, a
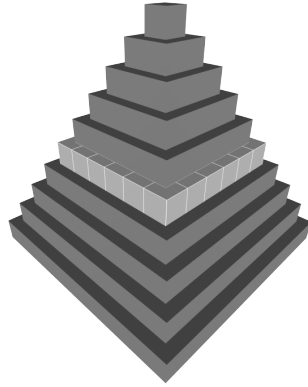
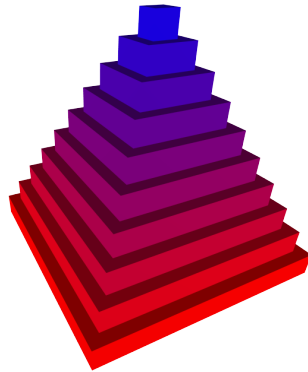Figure 3: A pyramid with ten levels composed of cubes

Figure 4: A pyramid with a different color for each level

material can be added to a node by attaching a `MaterialState` to the node. The material is defined by an ambient color, a diffuse color, a specular color, and a shininess coefficient.

```
var materialState = new MinSG.MaterialState();
// Color values are from the interval [0.0, 1.0]
var color = new Util.Color4f(red, green, blue, alpha);
materialState.setAmbient(color);
materialState.setDiffuse(color);
node.addState(materialState);
```

The final pyramid could look like the one shown in Figure 4.

General Hints

To execute a script inside PADrend, a very simple way is to store the script file in the `presets` directory in PADrend's main directory and then select it in PADrend using the SpeedDial-plugin (press `[F3]`). The script is evaluated whenever the button is pressed, so there is no need to restart PADrend after you changed the script.

To execute a single line of code, you can use the Console plugin (press `[^]`), which is bundled with PADrend. For example, to output information for the available attributes of a `MaterialState`, enter `info(MinSG.MaterialState);` and press "execute".