# **实现钢铁雄心那样的可点击地图（四）**

作者：[酸辣酸辣](http://gad.qq.com/user/index?id=677767)

链接：<http://gad.qq.com/article/detail/286566>

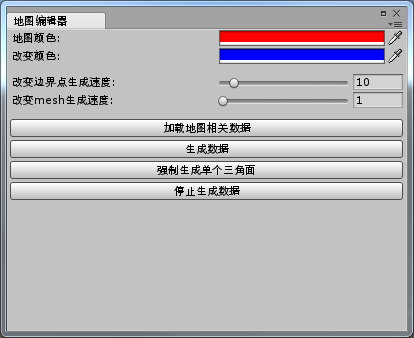
项目鸽了大半年，最近闲来无聊，便计划把原来挖的天坑慢慢填上。

然而Review代码时发现。。。代码写的真™丑

所以，在正式开发新功能前，插入这篇文章介绍代码的改动~

**一．**   **区分Edior模式和Game模式**

原项目中生成地图部分运行于Game模式下，虽然能用，但却没有Editor灵活，更重要的是——这样不**COOL**，为了让项目显得更**COOL**，决定将原来GenerateMap中的功能挪到MapEditor中在Editor模式运行，就像这样~



    添加了一些功能

1.  指定地图的颜色

GUILayout.BeginHorizontal();

EditorGUILayout.LabelField("地图颜色:");

MapData.MapColor = EditorGUILayout.ColorField(MapData.MapColor);

GUILayout.EndHorizontal();

GUILayout.BeginHorizontal();

EditorGUILayout.LabelField("改变颜色:");

MapData.ChangeColor = EditorGUILayout.ColorField(MapData.ChangeColor);

GUILayout.EndHorizontal();

2.  改变边界点的生成速度

GUILayout.BeginHorizontal();

EditorGUILayout.LabelField("改变边界点生成速度:");

MapData.BoarderGenerateSpeed = EditorGUILayout.Slider(MapData.BoarderGenerateSpeed, 1, 100);

GUILayout.EndHorizontal();

3.  改变mesh的生成速度

GUILayout.BeginHorizontal();

EditorGUILayout.LabelField("改变mesh生成速度:");

MapData.MapGenerateSpeed = EditorGUILayout.Slider(MapData.MapGenerateSpeed, 1, 10);

GUILayout.EndHorizontal();

4.  动态加载地图等~

if (GUILayout.Button("加载地图相关数据"))

｛

GameObject baseObj = GameObject.Find("BaseGameObject");

if (baseObj != null)

｛

GameObject.DestroyImmediate(baseObj);

｝

Object mapPrefab = Resources.Load("MapData/MapObject");

if (mapPrefab != null)

｛

mapObject = GameObject.Instantiate(mapPrefab) as GameObject;

mapObject.transform.parent = MapData.BaseGameObject.transform;

MapData.Image = null;

MapData.Image = (mapObject).GetComponentInChildren<RawImage>();

if (MapData.Image != null)

｛

Object mapTexture = Resources.Load("MapData/map");

if (mapTexture != null)

｛

if (MapData.Map != null)

｛

GameObject.DestroyImmediate(MapData.Map);

MapData.Map = null;

｝

MapData.Map = GameObject.Instantiate<Texture2D>(mapTexture as Texture2D);

mapSizeX = MapData.Map.width;

mapSizeY = MapData.Map.height;

MapData.Image.texture = MapData.Map;

MapData.Image.SetNativeSize();

isLoadData = true;

｝

mapTexture = Resources.Load("MapData/colorMap");

if (mapTexture != null)

｛

MapData.ColorMap = GameObject.Instantiate<Texture2D>(mapTexture as Texture2D);

｝

｝

｝

｝

**二．**   **减少代码减数据的耦合**

为了减少数据的耦合，将公共数据从GenerateMap中剥离，集中放到MapData中（然而并没有什么卵用），非公共数据下方到Province和Map中。

private static Color mapColor = Color.red;

private static Color changeColor = Color.blue;

private static Texture2D map = null;

private static Texture2D colorMap = null;

private static RawImage image;

public static bool IsForceContinue = false;

public static float BoarderGenerateSpeed = 10;

public static float MapGenerateSpeed = 1;

private static GameObject baseGameObject;

public static Color MapColor

｛

get

｛

return mapColor;

｝

set

｛

mapColor = value;

｝

｝

public static Color ChangeColor

｛

get

｛

return changeColor;

｝

set

｛

changeColor = value;

｝

｝

public static Color GetMapColor(int x, int y)

｛

return map.GetPixel(x - 1, y);

｝

public static Texture2D Map

｛

get

｛

return map;

｝

set

｛

if (map == null || value == null)

map = value;

｝

｝

public static Texture2D ColorMap

｛

get

｛

return colorMap;

｝

set

｛

if (colorMap == null || value == null)

colorMap = value;

｝

｝

public static RawImage Image

｛

get

｛

return image;

｝

set

｛

if (image == null || value == null)

image = value;

｝

｝

public static GameObject BaseGameObject

｛

get

｛

if (baseGameObject == null)

｛

baseGameObject = new GameObject();

baseGameObject.name = "BaseGameObject";

｝

return baseGameObject;

｝

｝

**三．**   **Mesh显示图片**

地图当然不能缺少颜色，使用一张带颜色的中国地图，在生成mesh时指定uv值，便能显示出地图本色~

List<Vector2> uvs = new List<Vector2>();

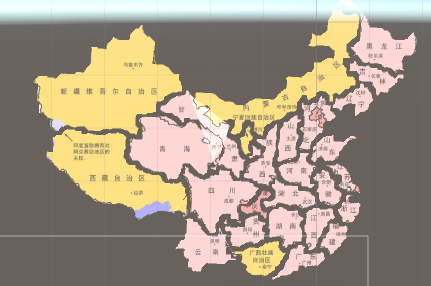
for(int i = 0; i < m\_vertices.Count; i++)

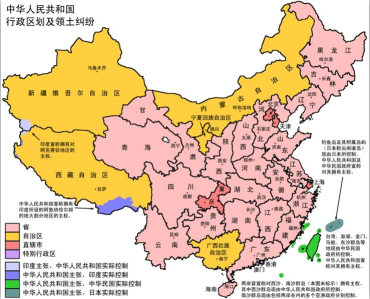
｛

uvs.Add(new Vector2(m\_vertices[i].x / MapData.Map.width, m\_vertices[i].y / MapData.Map.height));

｝

mesh.SetUVs(0,uvs);





最后附上完整的项目链接

<https://github.com/shangguanhun/MapGenerate>