# 一、测试环境

1.Windows操作系统，开发环境是win10专业版，

2.英伟达Nvidia GeForce GTX 1050显卡(2G显存)，同等级或者更好的显卡。

3.CPU是i5-6500 3.2GHz

4.16G内存

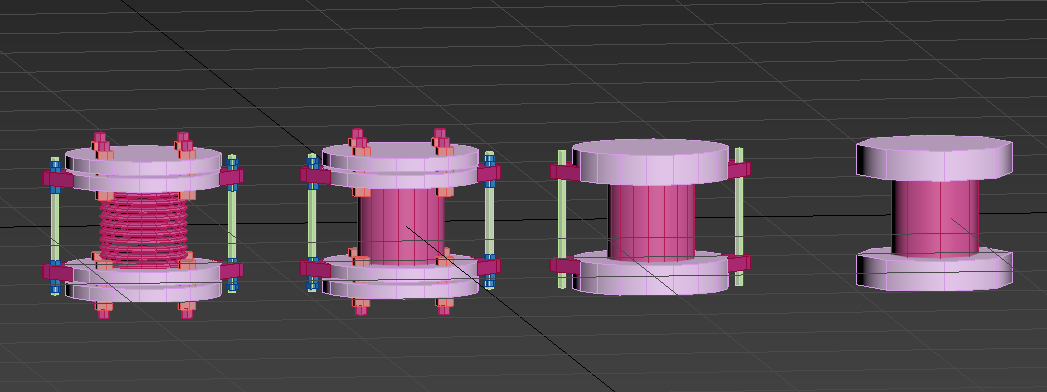
3.不需要安装运行环境

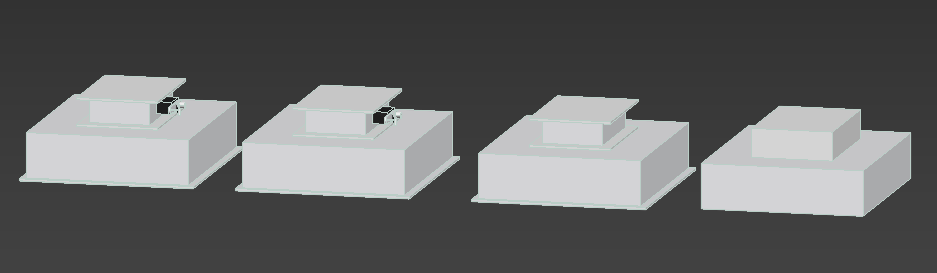
# 二、测试程序内容

1. 加载全部模型的性能测试
2. LOD处理模型的加载测试

# 三、模型处理

当前场景环境下，对Mesh点面占用最大的模型手动做了一下重建,这个模型点面数占比70%了。





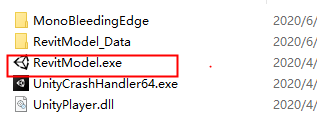
通过这几个模型的简化，将点面从8000w简化到了4000w。

其他模型用程序自动做了简化模型。

场景中，除了上述几个占比大的模型外，就是大量的桥架，桥架大部分无法复用，作为框架处理加载。

# 四、操作步骤

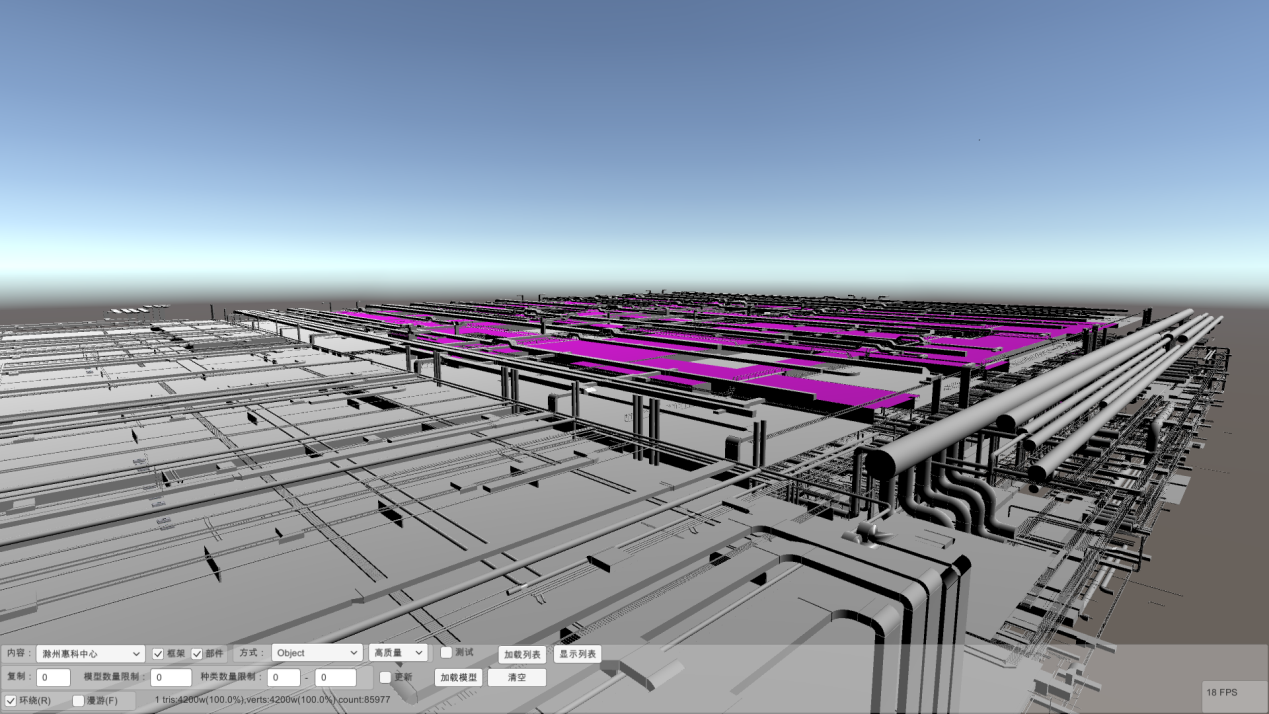
## 1.打开程序



## 2.点击加载模型



结果：



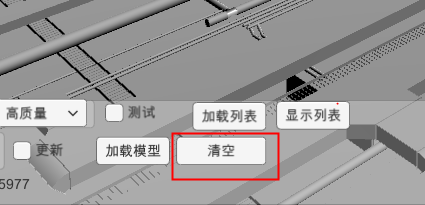
## 3.切换加载方式，测试运行效率，FPS多少。

提供了几种加载模型的方式用于选择和测试。



操作：

1)点击清空按钮，将当前模型环境删除



2)从方式下拉框选择一个其他加载方式

3)点击加载模型

不同方式比较：

**Object**：默认加载方式，直接加载原始模型。

测试帧率是10，但是操作上还能接受。

**Object(LOD)**：加载LOD处理后的模型，会根据距离切换模型的精细程度。

只多几种主要的模型做了LOD处理，其他大部分都没有做处理。在全部加载的情况下看不到明显帧率变化。在将大部分模型LOD处理完成后，运行效率会有所提高。

**Object2Entity**：加载好模型后，进行转换，将老的模型格式(GameObject)转换成新的模型格式(Entity)。不推荐，转换耗时。

**Entity**：新的模型渲染方式(EntityComponentSystem)，优点是充分利用现代GPU和CPU的特性，性能上比Object方式提高了将近一倍。

测试帧率是20，比Object更加流畅。不过加载时会比Object慢一点，需要做模型转换处理。

另外要利用新的渲染方式需要修改模型的材质，时间关系，并没有把全部的模型的材质设置完成，性能提高还是不够。

**Entity(LOD)**：Entity结合LOD，预期性能比Entity再提高一点。

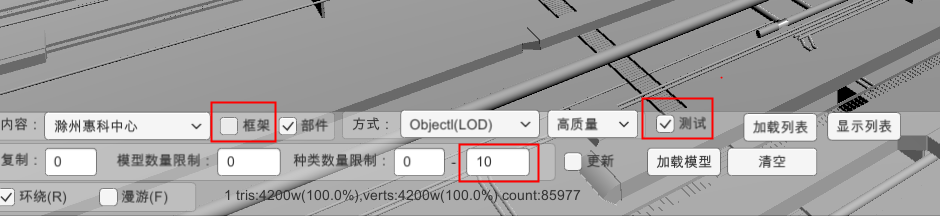
**Entity(Mesh)**：Entity的另一种创建方式，使用的API更加底层，比Entity方式性能稍微提高一点，同时会导致功能开发比较复杂一点。

1. LOD加载测试。

1)设置模型种类数量上限为10

2)去掉加载框架

3)勾上测试按钮

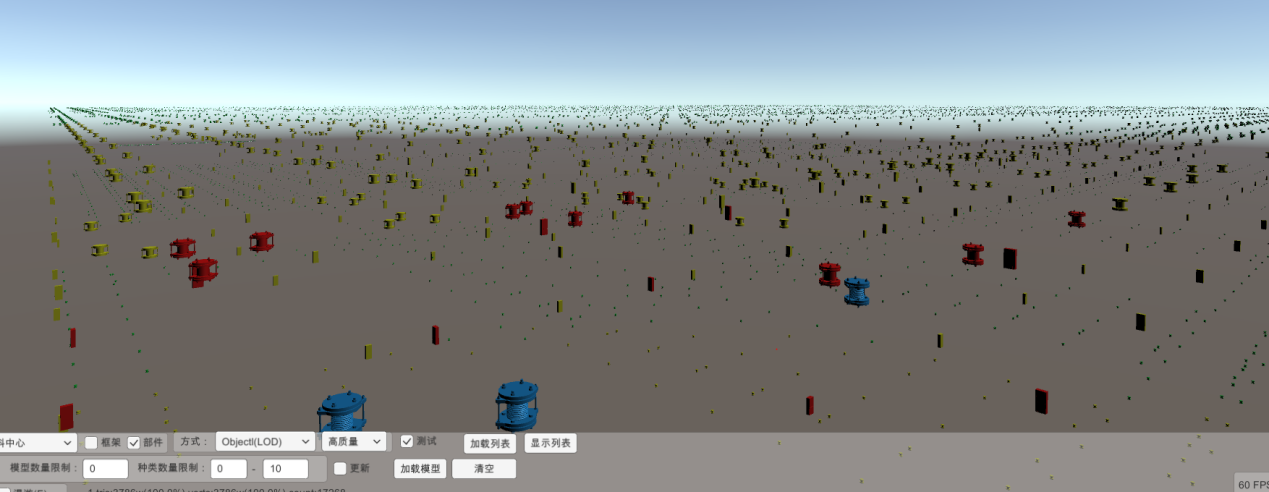


4)点击清空，选择Object(LOD)方式，点击加载模型

5)使用鼠标操作，不断切换拉近视角

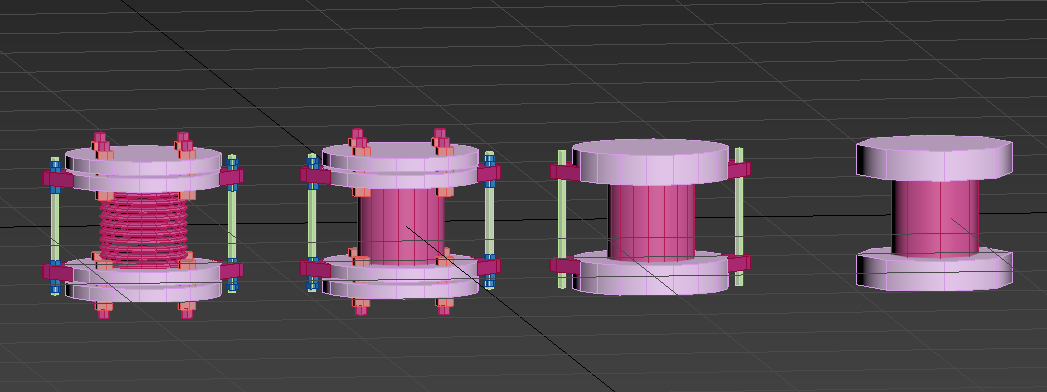
结果：

会看到做过LOD处理的模型会用不同的颜色标示精细程度

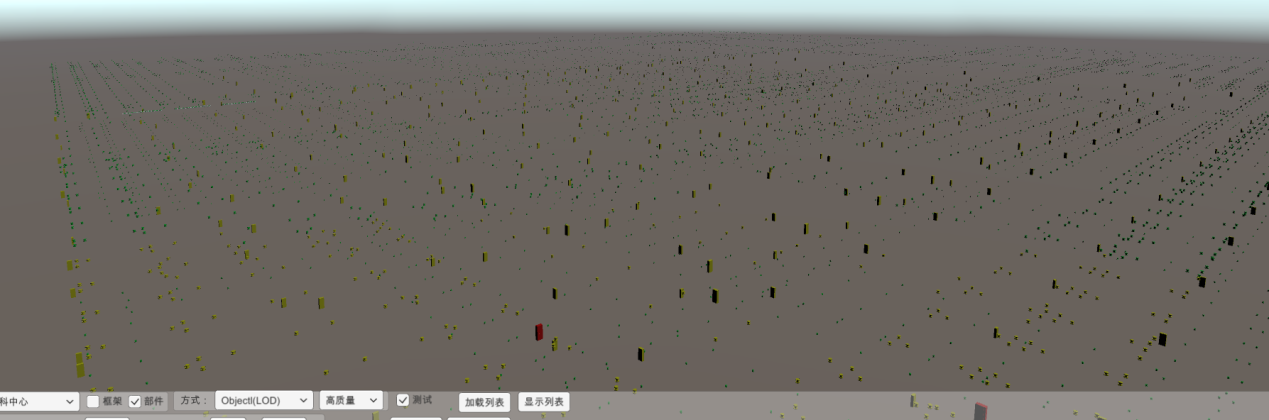


根据距离远近切换精细程度，不同精细程度分别设置为 蓝色->红色->黄色->绿色

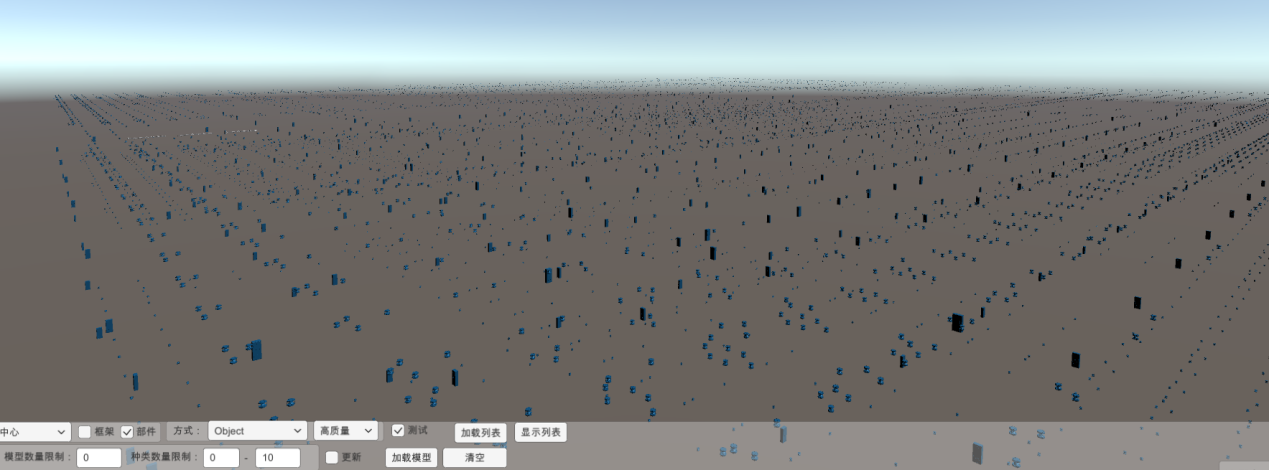
如下，模型从左到右依次是 蓝色->红色->黄色->绿色



可以看到大部分的模型是 黄色和绿色的



6)切换显示方式到GameObject进行比较



因为没有对所有的模型进行LOD和Entity设置，仅仅比较一下前10种的加载性能，点面占比是70%左右，2700w个点/面。

**GameObject： 34FPS**

**Entity: 44FPS**

**GameObject(LOD)： 48FPS**

**Entity(LOD)： 60FPS**

**需要了解的是，模型复用率越高，优化效率越高，后面的几千种复用率低的模型最终会拉低整个模型加载的帧率。**

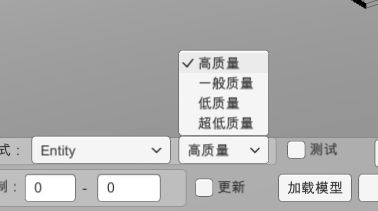
1. 按模型类别进行模型加载和卸载
2. 点击加载列表



1. 点击类型列表中的Load按钮，会独立加载该类型模型。加载后按钮变为Unload，可独立卸载该类型模型。
2. 点击类型列表中的复选框，可控制模型的显示和隐藏。
3. 逐个点击类型列表中的Load，观察帧率变化。

## 6．提供了单独加载不同精细度模型的选项

1)切换显示质量下拉框



2)清空->加载模型

3)查看不同质量条件下的点面数量和帧率

随着模型显示质量的下载，帧率会有所提高

# 五、结论

使用Entity渲染方式和LOD技术，能够有效保证程序的运行流畅度。

另外由于加载模型多，细节全部都加载了，桥架上的杆子、靠近的管道，会导致画面出现“闪烁”的效果，需要引入抗锯齿技术。这是显示方面的问题了。