

遇见非规则（下）： 聚沙成塔——粒子系统

华中科技大学软件学院 万琳



提纲

- ① 什么是粒子系统？
- ② 粒子系统实例

1

什么是粒子系统？

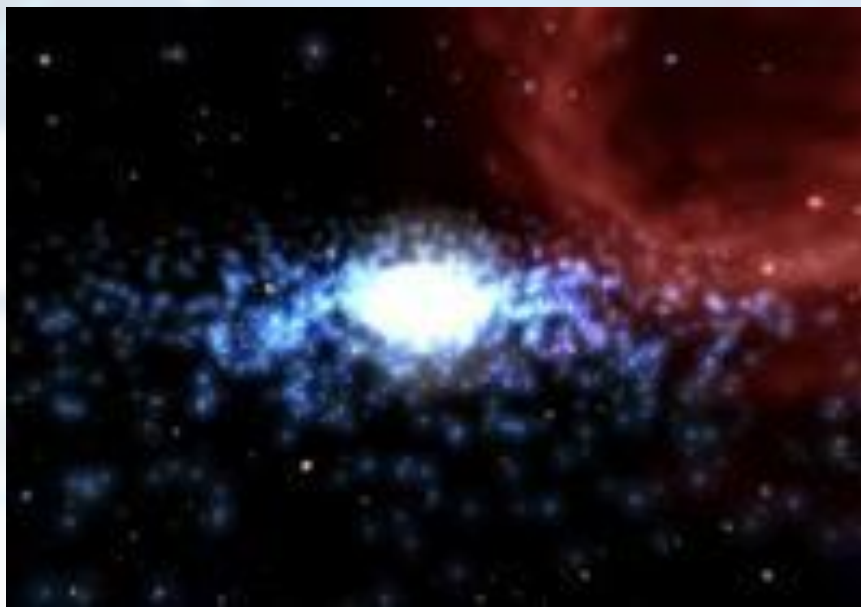
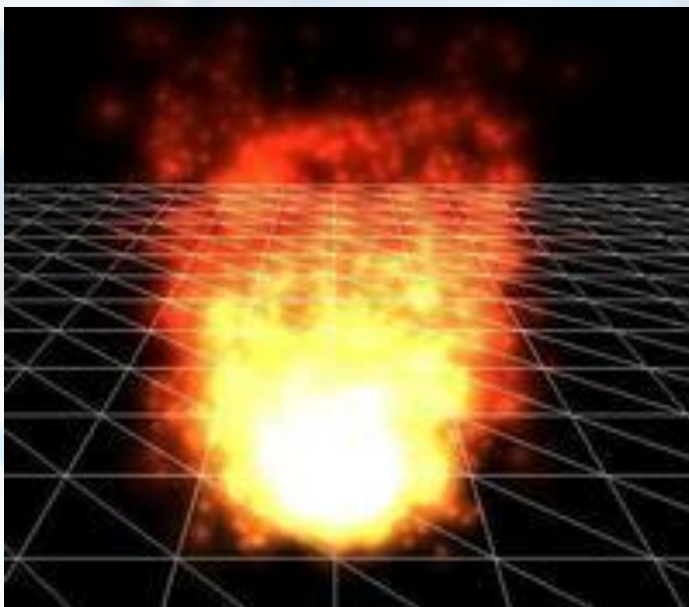
经常使用粒子系统模拟的现象有火、爆炸、烟、水流、火花、落叶、云、雾、雪、尘、流星尾迹或者象发光轨迹这样的抽象视觉效果等等。



1

什么是粒子系统？

经常使用粒子系统模拟的现象有火、爆炸、烟、水流、火花、落叶、云、雾、雪、尘、流星尾迹或者象发光轨迹这样的抽象视觉效果等等。



1

什么是粒子系统？

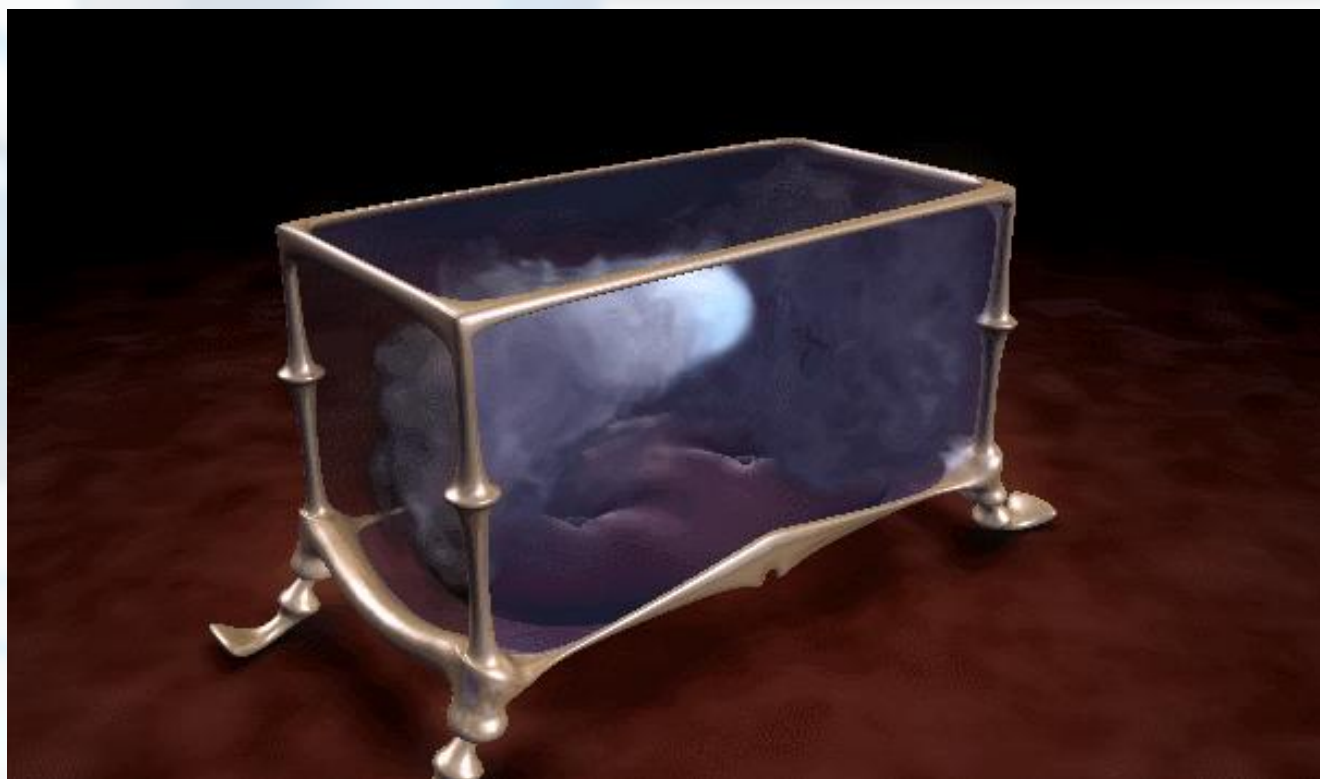
微粒系统 (particle systems) 是一组分散的微小物体集合，这些微小物体
的大小和形状可随时间变化，也就是说它们可以按照某种算法运动变化。

- 尤其擅长描述随时间变化的物体，可以用于模拟自然景物或模拟其它非规则形状物体展示“流体”性质。
- 微粒运动的模拟方式：随机过程模拟、运动路径模拟、力学模拟

1

什么是粒子系统？

展示“流体”性质：



1

什么是粒子系统？

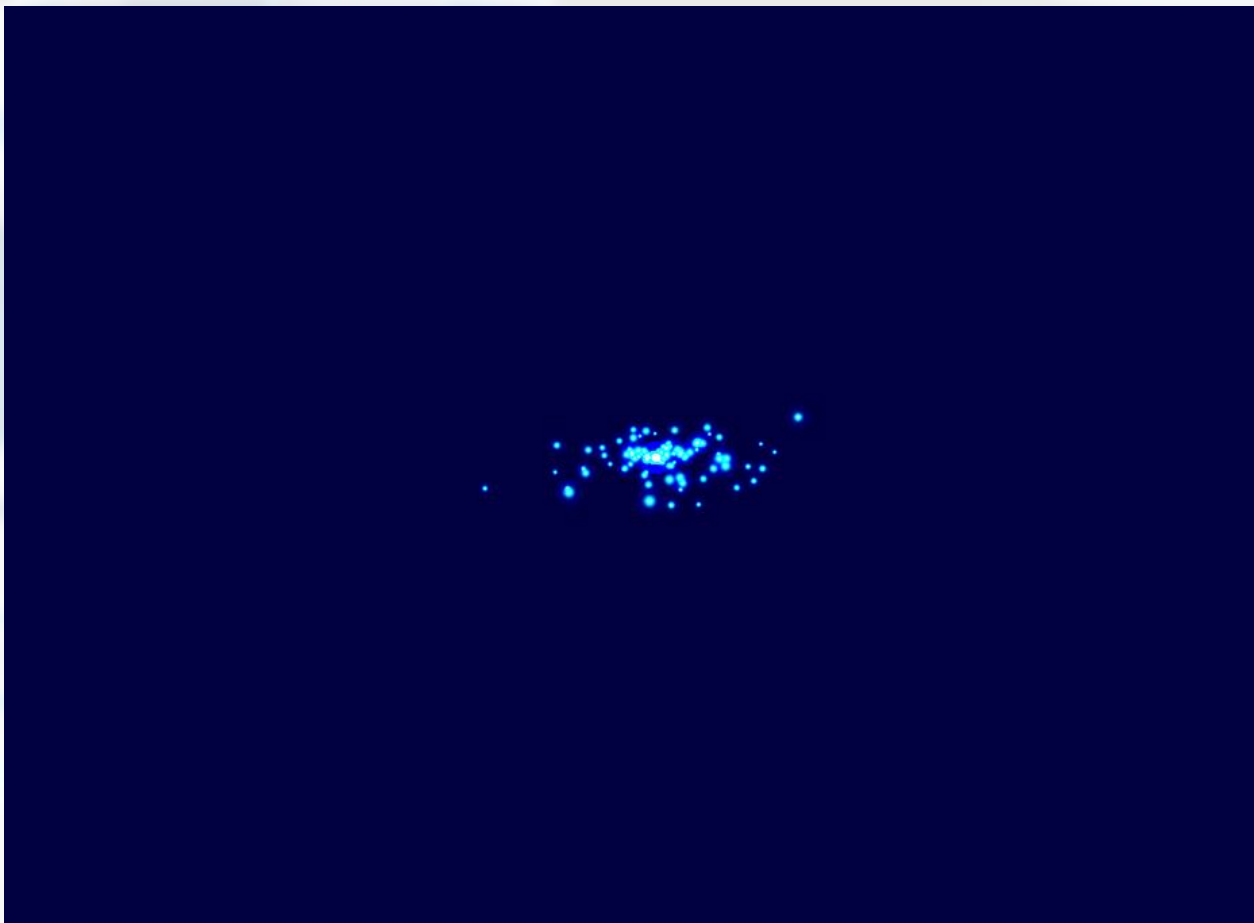
展示“流体”性质：



1

什么是粒子系统？

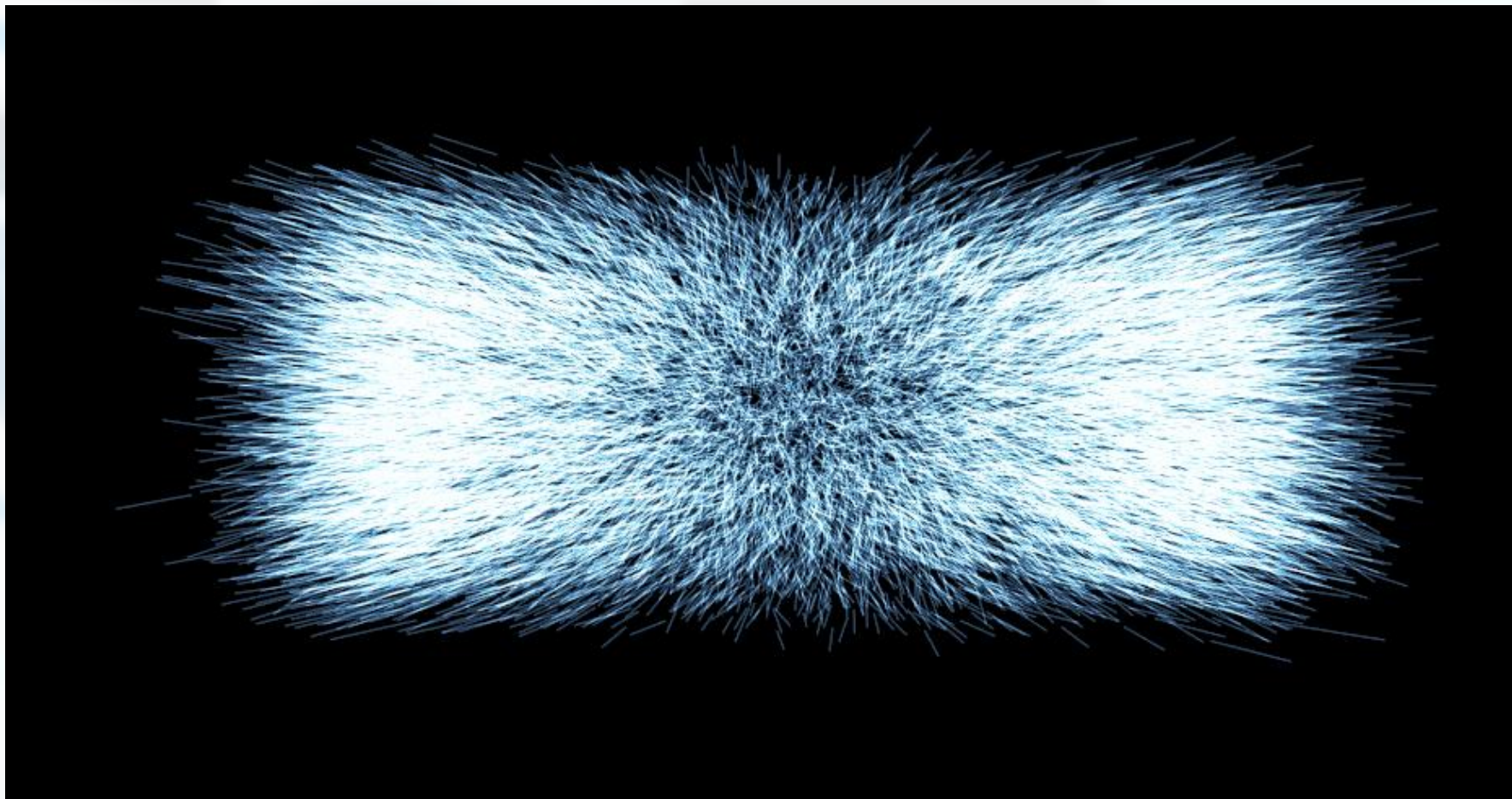
模拟不同的运动方式：



1

什么是粒子系统？

模拟不同的运动方式：



2

粒子系统实例

生成的两要素：

- ◆ 粒子本身的造型
- ◆ 粒子的运动方式

生成的两个过程：

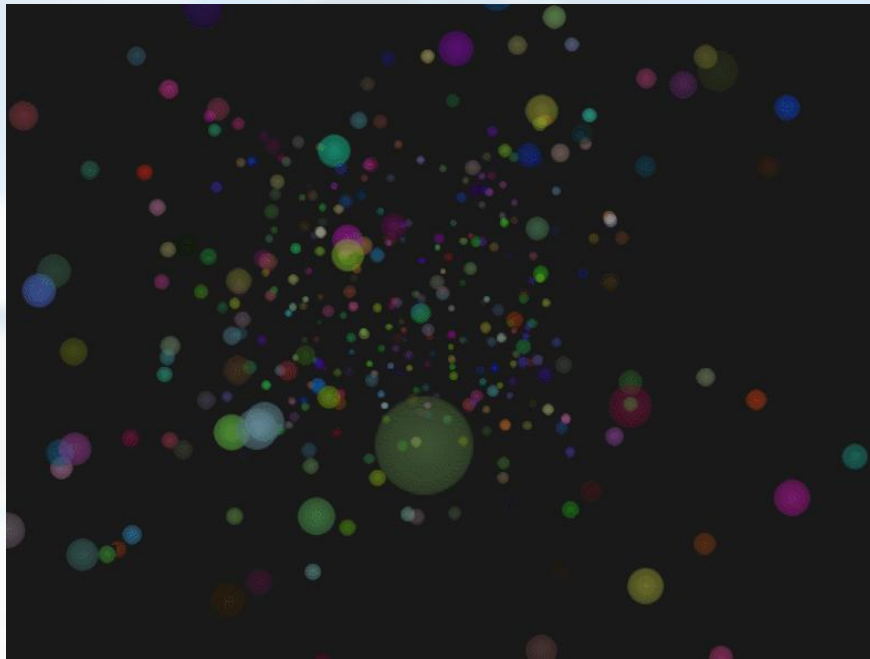
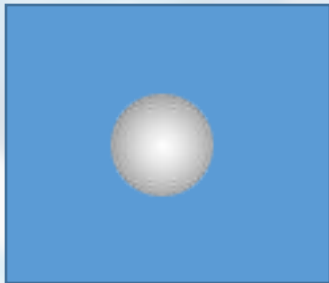
- ◆ 模拟
- ◆ 渲染

2

粒子系统实例

生成的两要素：

- ◆ 粒子本身的造型：圆形
- ◆ 粒子的运动方式：向屏幕方向运动



2

粒子系统实例

生成的两要素：

- ◆ 粒子本身的造型：雪花
- ◆ 粒子的运动方式：向上喷射，模拟重力落下



2

粒子系统实例

生成过程：模拟多个粒子及其运动

显然，单一的粒子没有什么效果。要实现这种粒子效果，需要持续不断的产生新的微粒，并且让旧的微粒随着时间逐渐消亡。

一个单一的粒子通常有一个生命值变量(通常是时间)，并且从它从产生开始就一直在缓慢的减少。一旦它的生命值少于某个极限值(通常是0)，我们就会“杀掉”这个粒子，用下一个粒子替换它。

一个粒子通常有下面的属性：

```
struct Particle
{
    glm::vec3 pos, speed; // 粒子的位置、速度
    unsigned char r, g, b, a; // 颜色
    float size; // 粒子大小
    float life; // 粒子的剩余生命，小于0表示消亡.
}
```

2

粒子系统实例

生成过程：模拟多个粒子及其运动

我们可以定义一个粒子数组 ParticlesContainer

```
const int MaxParticles = 500; //最大粒子数  
Particle ParticlesContainer[MaxParticles]; //粒子数组
```


2

粒子系统实例

生成过程：模拟多个粒子及其运动

在绘制循环中，可以计算出每一帧的时间差值 **deltaTime**，根据这个差值可以计算粒子剩余的生命、以及粒子在这段时间的位置变化

```
lastFrame = glfwGetTime();
while (!glfwWindowShouldClose(window))
{
    float currentFrame = glfwGetTime();
    deltaTime = currentFrame - lastFrame;
    lastFrame = currentFrame;
}
```

2

粒子系统实例

生成过程：模拟多个粒子及其运动

根据上面计算出的 **deltaTime**，更新粒子数组中粒子的生命和位置：

```
for (int i = 0; i < MaxParticles; i++)  
{  
    Particle& p = ParticlesContainer[i];  
    if (p.life > 0.0f)  
    {  
        p.life -= deltaTime;  
        if (p.life > 0.0f) p.pos += p.speed * (float)deltaTime;  
    }  
}
```

同时，需要 new 出新的粒子，替换掉粒子数组中已经死亡（如life小于0）的粒子。

2

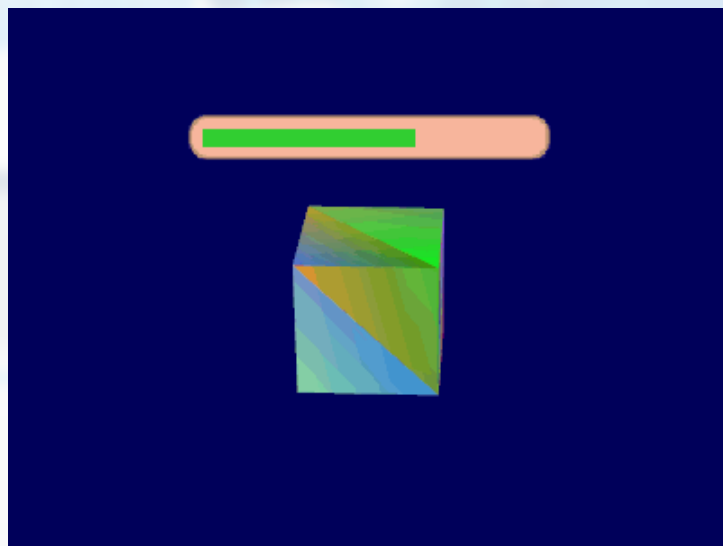
粒子系统实例

渲染过程：公告板技术

粒子始终面向摄像机方向，需要用到公告板（ Billboards ）技术。

公告板技术，形象的来说，就好像一个人举着牌子，无论你从哪个方向看向牌子，那个人都会把牌子朝着你的方向旋转，你永远只能看到牌子的正面。

如3D游戏中的血条和NPC上的名字（ Non-Player Character不受玩家操纵的游戏角色 ）。

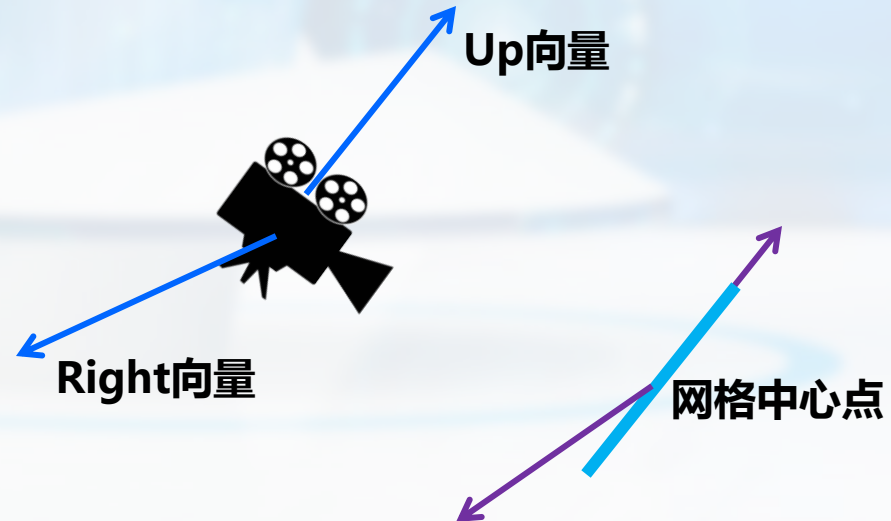


2

粒子系统实例

渲染过程：公告板技术

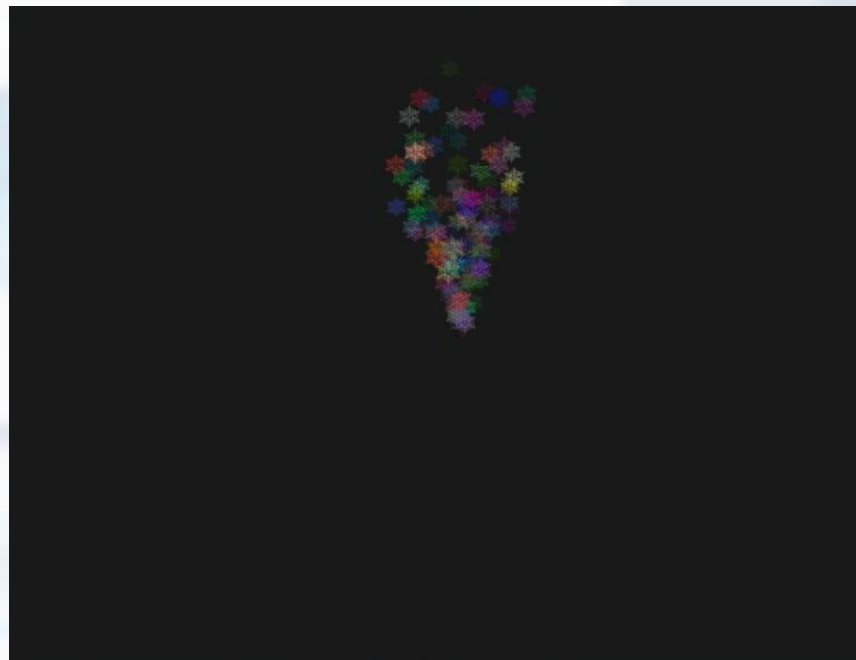
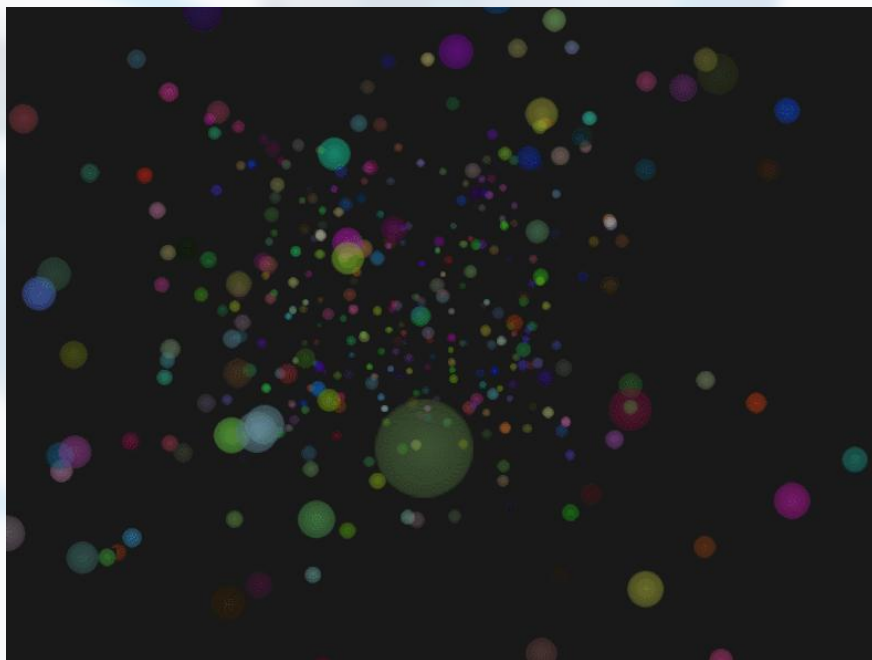
我们可以将公告板技术应用到粒子的渲染中。根据粒子中心在世界空间的坐标，以及观察者（摄像机）的位置，算出粒子四个顶点的坐标，从而让这个粒子始终面向观察者（摄像机）。



2

粒子系统实例

例程：





谢谢

软件学院 万琳