# 搜狐畅游------游戏引擎开发

1. 有一些房子，建筑物，树，树叶，草，天空盒，地面，水，如何保证渲染效果正确？

天空盒为什么最先绘制？

有几种绘制方式？ 哪种更好？

如果先绘制，后续不透明物体的渲染如何保证能够将天空盒效果混合到不透明物体上？

不透明的渲染时，z-test开启还是关闭？Z-write开启还是关闭？

答：半透明物体的渲染，需要进行深度测试z-test，判断能否被看到， 但是不必进行zwrite，否则半透明物体后面的不透明物体看上去就会被遮挡

1. 什么阶段进行背面剔除
2. 怎么进行IBL
3. 说一下延迟渲染、forward+

场景中有许多点光源，有许多探照灯， 怎么做shading？ 保证每个光源照射到的范围不同。

目的：延迟渲染如何实现点光源、聚光灯效果的光源绘制。

利用衰减函数，计算出每个点光源的半径，通过计算该片段到光源的距离，比较小于光源半径就绘制。

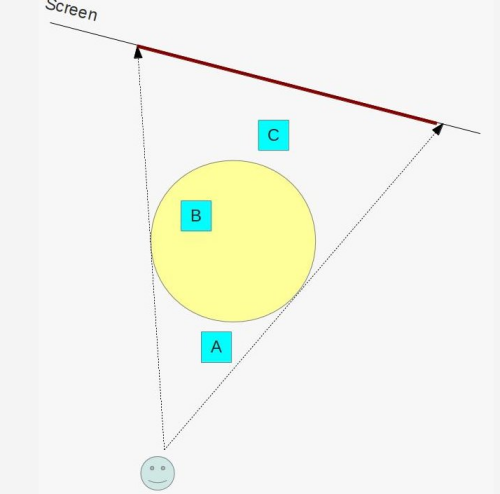
绘制效率不变，还是将所有的像素都绘制了一边。 SIMT架构会锁步执行。

改进：通过渲染一个衰减半径r的球体，来实现效果。

问题1：渲染是否开启背面剔除？ 如果开启，当相机进入光球中，此时将看不到球背面的东西。 如果关闭，当相机在球体前面，会对球正面、反面都绘制一次。

问题2：对于屏幕空间的物体，无法判断在世界坐标中其是否在球体覆盖范围里，有可能在前面，或者后面。

方法，使用模板缓冲



从相机角度观察时，球体的背面和正面多边形都在物体A的后面，且都在C的前面。

对于B(球体内部的物体)，正面多边形在物体B的前面，背面多边形在物体B的后面

先渲染场景，获得深度贴图，然后渲染球体，将深度缓冲只读，禁止背面剔除，设置模板缓冲写。 并对球体渲染两次，并设置相应的模板缓冲规则

模板缓冲规则：

渲染背面多边形时，如果没通过深度测试，就将模板缓冲的值+1

渲染正面多边形时，如果没通过深度测试，就将模板缓冲的值-1

那么，对于A，此时球体的正面和反面都通不过深度测试，+1，-1抵消

对于C，球体的都通过深度测试，不变

对于B，正面通过深度测试，反面没通过，+1

最后，我们得到模板缓冲的值，只需要将模板缓冲中不为0的部分，绘制即可！！！

并且开启正面剔除，可以少绘制一次，且进入光源不会出错。

1. 说一下CSM
2. 使用什么抓帧工具？ 抓过手机端吗？

如果一个场景，很多玩家，造成卡顿、手机发热，如何去优化？

说一说别的性能优化方式

1. 说一下动画，骨骼
2. 后续学习。。后处理效果，特效

角色效果，皮肤，颗粒，眼球，头发？

环境，海水的模拟，反射SSR，顶点起伏

植被，大面积起伏，跟人的交互，风力的交互

天气效果，雨雪，天气系统，24小时天气变换

后处理操作，Bloom，SSAO，TAA等

研究商业引擎

# 2、不鸣科技--工业引擎开发

一面：c++的项目问题，一个字，菜，准备的不够。 不光需要知其然，所以然，还需要进一步了解如何优化的方式，学习市面上的效率高的容器算法，为什么效率高。

1. 说一说类模板和函数模板区别
2. 、类模板有特化和偏特化(T&, T&&, T\*, const T\*)，函数模板只有特化
3. 、类模板需要手动指定参数类型，函数模板可以自行推导
4. 、类模板可以设置默认参数，函数模板没有
5. 内存池部分

如何增加运行效率和减少内存碎片？ 如何量化这个标准跟市面上的内存池进行比较？

答：量化，通过vector，每次去push\_back，pop\_back，从内存中申请值，然后归还，设置一定数量。

如何通过给定大小，**去找到相应的指针数组下标？** 比如给定7，需要找到第一个指针(存储8字节的指针)

二分，他说不行。

本质是位运算，假设指针数组指向的内存大小分别是8,16,24,32。。。

1-8 调用0号，8-15调用1号。 通过 (bytes + (8 - 1) ) / 8 - 1

补充点，**如何将bytes上调整到8的最小倍数**

1-8调整到8,9-16调整到16. 通过 (bytes + (8 - 1)) & ~(8 - 1) , 8到15 与上11111000

**为什么选择链表来管理内存**

答：考虑到头部插入和删除时间比较快。

1. Priority\_queue

实现过程中，有什么比标准库更优化的地方？

答：在向下调整的时候，STL中的实现方式时，记录根节点的值，将左右节点的较大的值直接移到根节点，然后递归往下处理，然后在回溯的时候将原先记录的值跟每个根节点比较，如果更大就将原本上移的再移回来，直到记录的值小于根节点的值。 对于完全有序的情况下，会重复操作，先全部上移，再下移。 可以设置一个阈值，当树高小于一定程度，直接使用stl的方式，当树过高，可以先进行比较再移动，这样如果完全有序，就不会冗余操作。

**什么地方使用到了优先队列？**

Topk，定时器

1. Unordered\_map

如何实现，有什么不一样的地方？

在扩容的时候，当数据量过大的时候，选择一种慢迁移的方式，不然一次插入，可能导致数据迁移时间过长，等待过长，将后续每次插入都选择旧容器中的部分数据拷贝到新元素中，代价是后续的查找需要在两次表中分开查找，将一次的代价平均到几次操作中。

**怎么样选择装填因子？**

**答：一般选择0.75， 经验值，如果是0.5就会造成剩余空间容量的浪费，如果过大，会造成数据冲突几率增加。**

**扩容后需要将数据拷贝过去？**

**答：不需要拷贝，对于每个节点直接获取到在新容器的下标，然后将节点移动过去，改变指针就行。**

**如何解决哈希冲突？**

**答：使用拉链法，后续通过AVL或者红黑树进行优化。**

除此之外呢？

**跳表的方式，管理若干级索引，从上至下，每个索引管理其中几个节点，查找的时候从上往下找，当找到后一个节点的值比当前大，就往下一级查找，直到相等。 时间复杂度logn； (如何管理插入，通过从下往上，第一级一定插入，后续每1/2的概率插入节点，这样第i层就是1/2^i，保证上一层是下一层的1/2)**

**可以用跳表吗？如果不能，为什么**

**答：可以(感觉是不可以的样子)**

1. Shared\_ptr

实现过程中，如果如何解决循环引用？

答：加入一个weak\_ptr,可以指向shared\_ptr指向的对象，但是不会增加引用计数，后续使用lock返回shared\_ptr对象。通过对象是否为空来判断是否被析构了。

实现过程中Lock返回的时候如何判断指针对象是否在其他地方被释放？

答：lock的底层实现，返回一个shared\_ptr, 通过构造局部变量，空的shared\_ptr,调用一个从weak\_ptr接受指针的函数，函数里判断如果计数器不为0，返回true，并将shared\_ptr的指针指向weakptr， 如果返回false，什么都不做。 最后将这个shared\_ptr返回即可。

**智能指针是线程安全的吗？**

**答：不是**

1. 线程池

如何实现？

1. 任务，task。 Function + 可变参模板
2. 任务队列， 生产者消费者模型
3. 哈希映射，线程ID，bool，来开启和终止线程
4. 通过mutex和条件变量实现互斥

Function在指向lambda表达式时，lambda捕获的对象此时可能被析构的，此时运行function就会出错，如何解决？

答：try catch。 然后呢？

通过改为值捕获，或者在捕获某个this对象时，将数据拷贝一份局部变量的值，通过捕获这个拷贝值。

**条件变量的虚假唤醒怎么解决？**

答：通过while不断地判断条件是否得到满足。本质是多个线程进去阻塞队列时，此时生产者发送notify唤醒，然后阻塞队列中的线程得到锁，去队列中取数据时，发现数据已经被别的取走，导致出错。

**Notify的信号丢失问题，怎么解决？**

答：首先，可能出现先notify，然后再wait的情况，出现信号丢失。

使用一个全局变量，一个线程在发送信号后，变成true，另一个线程进入wait之前判断，如果是true直接继续执行就行，不用去等待。

# 3、快手图形算法

一面：

1、deque的实现方式

每个缓冲区多大

1. 实现vector中的有意思的点

如何实现，push\_back，扩容机制，几倍扩容，实现构造和拷贝使用原地构造，赋值运算符需要先析构原来的内存

1. 模板的使用， 写一个min函数

有什么需要注意的地方

传入const T&, 返回const T&

如果不是没加const，不支持min(3, 4)， 因为T &a = 3 错误， 但是const T &a = 3 没问题

template<typename T>

const T& *min*(const T& a, const T& b) {

if (a < b) return a;

return b;

}

此外，对于T类需要重载小于号运算符，怎么重载

1. 讲一讲线程池的实现

通过什么实现并发互斥， 上锁、条件变量

**遇到过什么bug？**

**Function绑定任务的时候，如果对象被销毁，怎么解决，如果指向一个野指针？**

**使用智能指针？ 不一定有用，使用前判断下这个对象是否还存在**

1. **type\_traits**
2. **延迟渲染**

**Forward+实现，2维的缺点，深度信息问题， 转成cluster，三维的问题，数量太大，如何优化？ 对于远处的块划分的区域大一点，近处的小一点，因为是透视投影。 深度缓冲剔除一些看不见的**

1. **shadowmap 遇到的问题**

**自遮挡怎么解决？ 加一个偏移量， 直接加吗？ 通过往法线方向加便宜**

1. **对于技能特效较多，场景卡顿，如何优化？**

**特效的材质？ 半透明， drawcall有影响吗？ 不会，因为一般合成一个batch**

**答：从内存管理的角度，很多资源的加载，对象的生产销毁，从渲染的角度，半透明材质渲染耗时**

**因为场景中第一次加载不会把技能特效加载进去，在使用的时候才会加载，并且第一次加载进来以后，第二次使用就不会很卡， 如何优化？**

**通过预加载？ 某几个角色出现在你身边，就预加载进来，比如moba游戏 10个英雄，将英雄加载**

**如果都加载进来，内存不够怎么办？ 将距离远的替换，或者LRU。。 那就写个LRU**

二面：

1、天空盒的实现，给定向量如何采样

V矩阵如何变化，P矩阵如何变化(xyww)？

1. PBR讲一下， IBL
2. GI部分，很少使用IBL，太慢，有了解其他的部分吗？比如球谐函数，lightmap，SSR

答：讲解球谐函数如何使用，SSR使用以及缺点

问：SSR步进的时候，步长怎么确定？ 如果太长了就可能采样点少，如何太短了就太慢。。

答：不知道

1. TBN矩阵原理，实现的两种方式

再问：vs(切线空间)中计算一定会更快吗？

答：不一定，可能三角形很小，vs数量和fs差不多

1. early-z 以及失效， 使用drawcall会变多吗？ 答：会多，计算两遍，
2. Overdraw了解吗？ 优化呢
3. CSM实现
4. BVH如何构建，如何优化？ SAH

再问：场景中的物体一定是小于2个物体才停止切分？

答：不一定，看场景，如果复杂就自定义数量

问：为什么不用八叉树，BSP等，而是用BVH

1. 使用过Instance吗？

答：在绘制重复的物体，比如草，只需要调用一次drawcall，就能在不同的地方绘制物体

10、不透明和透明物体的渲染

11、shader中if-else分支的影响

再问：一定有影响吗？GPU会优化吗？

# 4、B站 游戏算法岗

1、vs 和 fs都做了什么？

2、进入fs的最小单位是每个像素吗？

答：当时没太听懂。 后面他说是对每个三角形进行处理。。

1. earlyz的做法， 效率一定会更高吗？

答：做两个pass，第一个pass记录最近深度，第二个再深度相同渲染，不一定会更高，大多数情况下会更快

1. 讲一下光线追踪器的实现
2. 蒙特卡洛的原理，如何实现渲染方程？
3. 追问，如果给你一个fx = x，求01上的积分，如何使用蒙特卡洛采样？ 如果实现重要性采样呢？

7、TAA的实现

# 灵犀互娱



1. 说一下Static
2. c和c++的相互引用

答：extern 问：有了解原理吗

1. 多态
2. 菱形继承
3. New了一个指针再delete掉，有什么影响

答：普通的指针会未定义，类指针在调用成员方法时，对于没有修改成员变量的方法可以访问，因为成员变量在堆上的内存被释放了，但是成员方法在代码区。

1. 渲染管线
2. 金属工作流
3. Shadowmap
4. 延迟渲染和正向渲染
5. 完美一面
6. 软光栅渲染器实现过程中，印象比较深刻的

答：shadowmap 和 IBL

1. 光线追踪器的BVH构建 以及优化， 追问：移动物体如何使用BVH

答：SAH

1. 虚函数
2. STL的sort

# 6、剑心互娱

1、软光栅如何实现MipMap?

2、各种抗锯齿？

TAA在镜头不动时，上下两帧的同一个点颜色一样吗？

1. 如何推导骨骼动画？ 一个顶点在局部空间，如何转化到当前姿势模型坐标
2. 虚函数， 构造函数里调用虚函数，可以调用吗？ 会报错吗？

答：不会报错，但是只能调用到基类的虚函数。 不推荐这种写法。

1. 模板的成员函数可以定义成虚函数吗？

答：不能

1. c++源文件到执行文件的过程， 头文件中声明但是没有定义，在另一个文件中调用，会出错吗？ 哪个阶段出错？ 如果两个文件中定义相同的变量，比如int，会报错吗？ 哪个阶段？
2. 对相同长度的数组和链表 遍历哪个效率更高？
3. 右值引用的作用？
4. c++的模板使用过程中可以.h和.cpp分离吗？ 如果不能，为什么，如何实现分离？