0，不停重构。

1，考虑用插件式，微内核。先实现一个小的C++插件系统，学习下x3c，https://sourceforge.net/projects/x3c/，他的博客在https://www.cnblogs.com/rhcad/category/254079.html

2,core部分包含哪些。生成.dll哪些

1. 看看鬼火引擎的各部分划分编排方式
2. 尽可能完成shader接口，尽快加入shader，其余的可以慢慢调
3. 分层的思想。将每个绘制图元都看做一个层。
4. Qt用单文档菜单列表格式。
5. 核心编程，看看如何加载各个dll
6. 用boost或者stl
7. 随时加进去可以加的东西。比如gis
8. 加载配置文件的方式来初始化布局，用以动态增删相应的内容
9. 添加一个图元基类。通过map绘制这些图元。
10. 将算法和算法ui写成dll，动态加载和卸载。
11. 关联关系：UI->上下文层->绘制层。
12. 每帧绘制的是当前层的图元，当前层是所有绘制层中的一个，通过动态调度来选择当前层，增删上一层与当前层不同的元素。开个线程来判断是否不同。
13. 内存管理。
14. Cocos每个类都会有个帧循环的函数，用的监听者模式