1. 线程安全的对象生命期管理
2. Observer（观察者）设计模式

观察者模式定义了一种一对多的依赖关系，让多个观察者对象同时监听某一个主体对象。这个主体对象在发生状态变化时，会通知所有观察者对象，使他们能够自动更新自己。  
一个软件系统常常要求某一个对象的状态发生变化的时候，某些其他的对象作出相应的改变。做到这一点的设计方案有很多，但是为了使系统能够易于复用，应该选择低耦合度的设计方案。减少对象之间的耦合有利于系统的复用，但是同时设计师需要使这些低耦合度的对象之间能够维持行动的协调一致，保证高度的协作（Collaboration）。观察者模式是满足这一要求的各种设计方案中最重要的一种。



从上面的解释可以看出，观察者模式是非常适合客户-服务器模式的编程的。

1. 析构函数遇到多线程会存在的问题：

第二章 线程同步精要

第三章 多线程服务器的使用场合与常用编程模型

第四章 C++多线程系统编程精要

第五章 高效的多线程日志