# 背景

之前所述，select、poll、epoll等IO多路复用都是客户端连接服务器技术，线程池位于客户端，即客户端已经完成与服务端的连接，可以使用高效地方法接收数据，接下来就是如何高效地处理数据。

# 原理

提前创建好一堆线程，客户端连接的时候直接使用，不需要频繁地创建和销毁线程，提高响应客户端的时间。

客户端发过来的数据对应一个事件，在服务端维护着事件的任务队列（这里使用条件变量判断是否需要启动线程相应事件，即任务队列不为空时线程池调用pthread\_sinal或pthread\_brodcast取任务，否则调用pthread\_cond\_wait阻塞），客户端发过来的事件在服务端会启动相应的线程处理。

# 代码

# 应用