<https://blog.csdn.net/weiyuefei/article/details/52242812>

分析了ET和LT的实现方式，那么分析他们的触发方式就容易多了。我们通过实现分析知道LT模式下epoll\_wait被唤醒可以通过两种方式（图中红线和蓝线），而ET模式只能通过一种方式（图中红线）。所以ET模式下能被唤醒的情况,LT模式下一定也能被唤醒。我们先来讨论特殊情况（ET模式）,再来讨论一般情况（LT模式）。

**2.1 ET**

根据上一节对两种加入rdlist途径的分析，可以得出ET模式下被唤醒（返回就绪）的条件为：

对于读取操作：

**(1) 当buffer由不可读状态变为可读的时候，即由空变为不空的时候。**

**(2) 当有新数据到达时，即buffer中的待读内容变多的时候。**

另外补充一点：

**(3) 当buffer中有数据可读（即buffer不空）且用户对相应fd进行epoll\_mod IN事件时**（具体见下节内容）。

对于情况(1)(2)分别对应图1(a),图1(b)。

对于写操作：

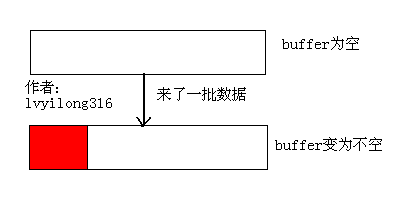
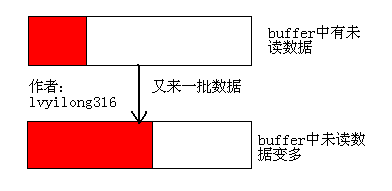
**(1) 当buffer由不可写变为可写的时候，即由满状态变为不满状态的时候。**

**(2) 当有旧数据被发送走时，即buffer中待写的内容变少得时候。**

另外补充一点：

**(3) 当buffer中有可写空间（即buffer不满）且用户对相应fd进行epoll\_mod OUT事件时**（具体见下节内容）。

对于情况(1)(2)分别对应图2(a),图2(b)。

                                    (a)                                                                                                       (b)

图1 ET读触发的两种情况

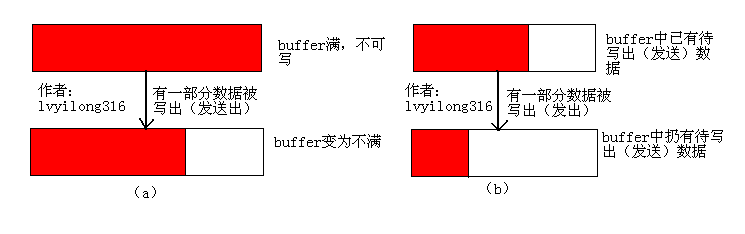


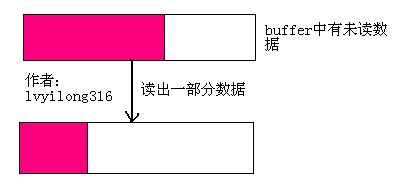
图2 LT写触发的两种情况

**2.2 LT**

LT模式下进程被唤醒（描述符就绪）的条件就简单多了，它包含ET模式的所有条件，也就是上述列出的六中读写被唤醒的条件都是用于LT模式。此外，还有更普通的情况LT可以被唤醒，而ET则不理会，这也是我们需要注意的情况。

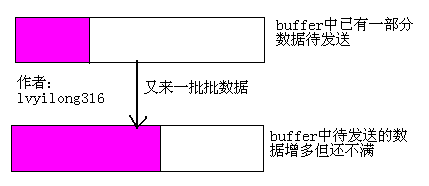
对于读操作

当buffer中有数据，且数据被读出一部分后buffer还不空的时候，即buffer中的内容减少的时候，LT模式返回读就绪。如下图所示。



对于写操作

当buffer不满，又写了一部分数据后扔然不满的的时候，即由于写操作的速度大于发送速度造成buffer中的内容增多的时候，LT模式会返回就绪。如下图所示。



注：poll和select都是LT模式。