**直播P2P协议**

LiveMS

LiveMS

LiveMS

LiveMS

LiveMS

LiveMS

**需求：**LiveMS可以设置source为上级LiveMS，不能设置同级LiveMS。

**多源备份**，规则如下：

1. LiveMS从N个source中随机选择一个，以防止所有的LiveMS都从同一个source下载；
2. 获取m3u8文件，解析m3u8文件，得到每个ts文件的生成时间；
   1. 如果最新的ts文件的生成时间<now-3\*10秒，那么该source无法支持服务，那么切换源点switch\_source()；
   2. 如果最新的ts文件的生成时间>=now-3\*10秒，那么该source可以继续使用。
3. 从一个source顺序下载每个ts文件；
   1. 如果ts已经下载，那么尝试下一个；
   2. 如果ts没有下载，那么下载；下载完成后，测试速率；
      1. 如果下载时间超过10秒，那么切换源点switch\_source()；如果偶尔一次超过，这个也正常，不应该马上切换；为简单起见，先实现为绝对的，后面实现平均时间不超过10秒。
      2. 如果下载时间在10秒内，那么该source可以继续使用；
4. 切换源点switch\_source()
   1. 如果只有一个source，无可切换，状态机不变，继续；
   2. 如果有多个source，那么切换到下一个source，
      1. 下一个source如果上次的下载时间也超过10秒，那么继续寻找下一个；
      2. 如果所有的source都超过10秒，那么寻找时间最小的source，在寻找过程中记下上次下载速率(bps)最高的一个。这是一个不稳定状态，永远都在切换。
5. 状态机有两个状态变量，请求m3u8文件(state\_m3u8)，和请求ts文件(state\_tss)。switch\_source时，将状态机记为state\_m3u8（因为每个source有自己的ts文件列表，每个source拥有的ts文件列表可能不同）。

LiveMS

LiveMS

LiveMS

LiveMS

LiveMS

LiveMS

**需求：**LiveMS可以设置source为上级LiveMS，也可以设置同级LiveMS。

**直播P2P**，规则如下：

1. 区分上级source，还是同级source，设置的时候要区分。
2. LiveMS选择所有的M个上级source，和N-M个同级source，作为peer。
3. LiveMS连接所有的N个peer，交换bitfield，
   1. 如果peer的拥有的ts文件过老（过老条件：在60\*10秒=10分钟以前），忽略；
   2. 按照ts文件的年龄，顺序选择前X个下载文件；（因为直播强调顺序性，优先下载了后面的文件，返回给普通播放器/非P2P播放器，并无用处）。
   3. 每个下载文件，随机选择其中的一个piece下载。
      1. 下载超时，换另外一个source，重试；
         1. 如果该source为同级source，记下失败次数；
         2. 如果失败累计三次，淘汰该peer，选择新的peer进来。
         3. 选择新的peer，优先选择未使用过的peer；
         4. 如果所有的peer都连接过，那么选择下载速率最快的一个。
         5. 切换多次，下载进度会严重滞后。（什么情况下放弃该文件下载？）
      2. 下载完成，随机选择下一个piece下载；直到整个文件下载完成；
      3. 然后，顺序选择下一个文件。

上述的行为，是LiveMS作为ActivePeer的行为。

1. LiveMS作为PassivePeer，下载到piece后，通知所有的对端。同时响应对端所有的下载请求。