# 基于类bitTorrent协议的p2p文件共享系统

## 项目简介：

P2P，即点对点网络（Peer-to-Peer Network），是指一种基于去中心化的计算机网络模型，它不依赖于传统的服务器或中心节点，而是将节点之间的计算能力、存储空间、带宽等资源进行共享和协作，实现分布式计算、数据传输和共享等功能。

众所周知，传统的C/S模式的文件传输中，任意一个客户端想要请求一个文件都需要向服务端发起请求然后由服务器返回文件资源。服务器频繁大量地向大量客户端接收和发送数据时，服务器负载就会提高，而文件传输又恰好是一个需要大量带宽资源的服务，这就使服务器的上载带宽成为整个系统的主要瓶颈，想要解决这个瓶颈就需要增加网络设备的性能。这必将对服务提供者带来相应的成本提高，类似的场景有许多像CDN和一些大型的视频网站也都使用了p2p的技术进行加速。

本系统利用了P2P技术，将传统的客户端分布式的文件资源能够实现相互传输、相互补全，相应的流量也转移了用户与用户之间。从而解决了传统的C/S模式中服务端带宽受限的瓶颈。

考虑到系统网络的节点个数局限性，本系统采用了中心化的P2P类型，即由一台或多台服务器节点设备（我们称之为tracker）对P2P网络文件资源进行调配，虽然采用了服务端，但是由于服务器只接受简单且数据量少的文件请求信息，大流量的文件传输仍由各个P2P节点之间进行从而减少服务器负担。

首先介绍一下该系统中的名词

1. tracker：p2p网络中的服务节点，对P2P网络文件资源进行调配。
2. torrent文件（种子文件）：种子文件中存有分享文件的元信息，包括大小，名称，和唯一的哈希值。
3. 磁力链接：包含了tracker的地址和tracker中相应的种子文件的哈希值。
4. peer：有共同torrent文件的节点。

//为提高性能，节点的文件将被分割成块，相应的块使用多线程进行传输，并且不同节点之间会尽量保证优先下载不同的分块以此来保证性能。//(不一定有)

系统共享的原理如下

1. 先有分享者发布torrent文件到tracker服务器
2. tracker服务器维护一个torrent文件和磁力链接的映射
3. 由下载者向tracker服务器发送磁力链接，返回peer节点信息和torrent文件
4. 对peer节点建立链接，根据torrent文件进行下载

何文龙：

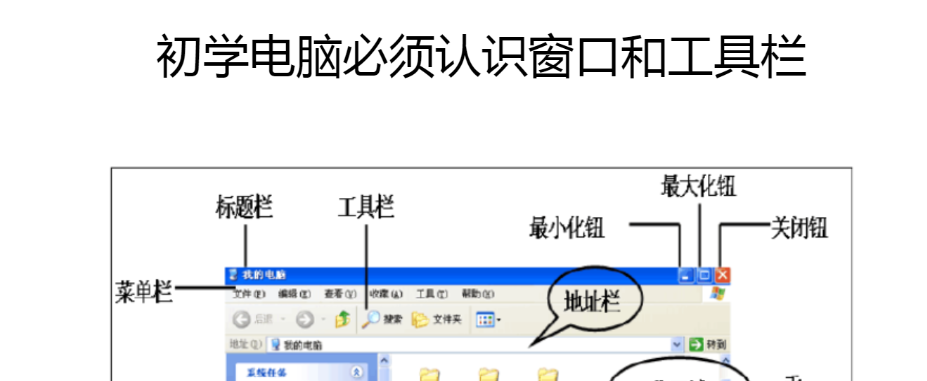
时泽中：

李茂源:

## 工程注意事项：

1. 注意命名规则尽量使用有意义的名称
2. 注意注释格式为 //内容+姓名+时间（可不加）

## 主要需求：

1. 用户端：
   1. 界面
      1. 
      2. 
      3. 标题栏随意
      4. 菜单栏
         1. 文件
            1. 制作torrent
            2. 从磁链下载
            3. 从torrent下载
         2. 帮助
         3. 相关
      5. 工具栏（可以带图标的按钮）
         1. 制作torrent
         2. 从磁链下载
         3. 从torrent下载（和菜单栏的一样，这三个按钮会打开新的小界面）
         4. 开始下载
         5. 停止下载
         6. 删除
         7. 打开目录
      6. 工具栏下（图）
         1. 用户界面
            1. 。。。
         2. 下载界面
         3. 详细页（保留）
   2. 登录
      1. 数据库
   3. 传输文件
      1. 传输前
         1. 检查本地文件夹中的torrent文件 √
         2. 获取torrent文件或磁力链接
         3. 向tracker拉取peers的ip地址
      2. 传输中
         1. 下载策略
            1. 贪心式
         2. 断点续传
         3. 监控流量
         4. 定时不断地和tracker进行通信传输peers的信息
         5. 不断地和peers交换下载和上传情况
         6. 分片上传和下载（保留）√
      3. 传输后
         1. 对文件分片的哈希值检验
   4. 制作torrent文件 √
2. tracker（服务端）
   1. 维护登录信息
   2. 界面（保留）
   3. 保存torrent文件
      1. torrent文件注册
   4. 根据peer对torrent文件的拥有情况对peer进行分类（定期）
   5. 向peer定期发送peer相关情况（由tracker？由peer？）
   6. 维护一个torrent文件和磁力链接的映射（hashtable？）
3. 分享页（不确定在客户端还是服务端，还是在客户端由客户端发送请求）