第2章 第一代P2P网络：混合式P2P体系

这一章讲述代表性的混合式 P2P 网络 Napster 和 BitTorrent。“混合式”在这里指的是 C/S 与 P2P 的混合，它反映了网络工作模式从 C/S 到 P2P 的过渡，Napster 正是如此。发展到后来的 BitTorrent 网络相当于分散化的多个 Napster 的集合，由此可见分布式的思想在混合式 P2P 网络里有着更深层次的渗透。

2.1 Napster——P2P 网络的先驱

Napster 是世界上第一个应用性 P2P 网络，也是混合式 P2P 体系最杰出的代表，它以不可抗拒的影响力向世界传达了 P2P 的独特思想，展现了 P2P 的巨大潜力，是 P2P 网络当之无愧的先驱。

2.1.1 Napster 出现的背景和它创造的奇迹

1999 年 18 岁的 Shawn Fanning 开发出 Napster，它提供服务允许音乐迷们交流 MP3 文件。与传统的提供音乐下载的网站不同的是，Napster 服务器里没有一首歌曲，它只是提供一个空间供音乐迷们将自己硬盘上的歌曲文件共享（用户发送歌曲索引信息到服务器）、搜索其他用户共享的歌曲文件

并到其他 Napster 用户硬盘上去下载歌曲。尽管 Napster 的工作原理听起来并不新奇，它所用到的软件技术也是计算机领域以前就有的，但是，它却是最先在互联网上让用户之间不经过服务器直接交换文件的应用体系。Napster 无意中采用了 P2P 模式、打破了客户/服务器模式的瓶颈，它背后隐藏的巨大的 P2P 世界也从此浮出水面。

Napster 发布后，在短短半年时间里吸引了 5000 万注册用户，最高时超过 6100 万用户，这在计算机网络领域是一个从未有过的奇迹。Napster 创造的奇迹同时也揭示了在互联网时代普通人也具有改变世界的能力——当 Shawn Fanning 最初在波士顿的东北大学校园开发 Napster 软件的时候，他只不过是想和弗吉尼亚的朋友共享 MP3 歌曲文件，而随后这个民间小软件影响了整个世界。

2.1.2　Napster 网络的工作原理

图 2.1.1 是 Napster 的工作原理图，Napster 网络由两个部分组成：Napster 网站＋Napster 用户。

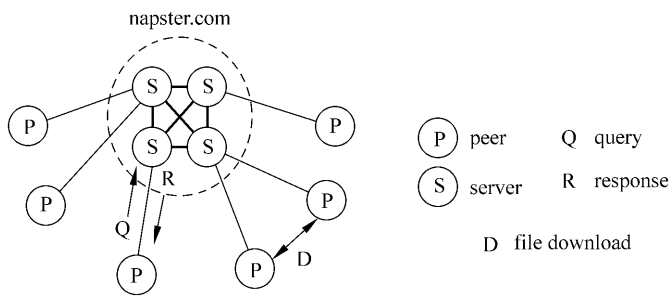

图 2.1.1　Napster 工作原理图
（来自[Saroiu et al.，2003]）

Napster 网站（图 2.1.1 中 napster.com）是一个服务器机群。每个服务器保存一部分用户的共享文件索引信息，所有的服务器互联、整合起来对网站外面的 Napster 用户提供统一的访问接口，在每个用户看来他们访问的都是同一个服务器。

每个 Napster 用户（图 2.1.1 中 peer）连接到机群中的一台服务器，他将愿意与其他用户共享的文件信息发送给服务器（图 2.1.1 中 server），服务器记录这些信息以及该用户的位置，并将它们做成一条索引添加到原有索引表中。当用户想要查询一个文件时，首先将“查询”消息发送给与其相连的服务器（图 2.1.1 中 Q：query），该服务器收到 Q 以后，与其他服务器协作处理查询消息 Q，处理完成后将“回复”消息（图 2.1.1 中 R：response）返回给用户，这条消息包含一个表单，列有所查到的所有匹配的文件索引。收到 R 以后，用户在表单中选择他想要的文件，根据文件索引中对应的位置与其他用户直接建立连接下载文件（图 2.1.1 中 D：file download）。

Napster 网站一方面维护所有 Napster 用户的共享文件索引，另一方面监控系统中每个用户的状态，比如跟踪记录用户所报告的连接带宽和用户已连入 Napster 网络的时间，以及发现哪些用户已经掉线等。这些信息对 Napster 系统的运作有重要的意义：首先，对于那些掉线的用户，必须在服务器中去掉该用户对应的文件索引，以保证文件索引的时效性；其次，当服务器收到用户的查询消息后，返回给该用户的回复消息中也包含了其他用户的连接带宽、连接时间等信息，这些信息对于该用户选择与哪些用户建立什么样的连接是很有帮助的。

2.1.3 Napster 的性能分析

文献[Saroiu et al.，2002；2003]和[SGG 02]对 Napster 做了详细的测量和分析，他们通过 Crawler(爬虫)工具来测量 Napster 网络。网络爬虫的工作原理大致如下[Sen and Wang，2004]："爬虫"首先作为一个结点加入 P2P 网络并与其他一些结点主动建立 TCP 连接，然后通过现有邻居获取更多结点信息、建立更多的连接，同时，"爬虫"结点记录下它与其他结点交换的信息以供后续分析。爬虫技术是"带宽密集型"的，因为它需要维持很多条 TCP 连接；同时，爬虫技术也是"侵犯型"的，因为爬虫自身参与到网络中来并对网络产生了影响，从而导致测量结果的不准确性(读者可以与物理学中著名的"测不准原理"相类比)。

根据他们的观察，Napster 网站机群由大约 160 台服务器组成，每个用户只和其中一台服务器建立连接，当这台服务器收到用户的查询消息时，它首先检索自己的索引以获得与它连接的所有"本地用户"的共享文件信息，然后检索其他服务器的索引以获得与其他服务器连接的"远程用户"的共享文件信息。

当新的用户加入 Napster 网络时，它告诉服务器自己的连接带宽，Napster 服务器记录这个值并向其他用户提供这个值，这对于系统的工作很有帮助，根据带宽信息用户可以选择与哪些用户建立什么样的连接。然而，许多 Napster 用户(大约 25%)加入网络时并不告诉服务器它的连接带宽，更坏的情况，许多用户故意告诉服务器比其实际带宽低很多的连接带宽，从而阻碍别的用户从它下载文件。

虽然 Napster 将所有的用户看成平等的成员，但是 Napster 用户之间在能力上却有着巨大的差异，也就是 P2P 所讲的结点"异构性"。就网络连接设施而言，约 25%的用户使用低带宽的调制解调器(Modem，64Kbps)，约 50%的用户使用较高带宽的连接(Cable，DSL，T1 或 T3)，约 20%的用户使用高带宽连接(至少 3Mbps)，详见图 2.1.2。就用户连接时间而言，超过 50%的用户连接时间低于 1h，不到 10%的用户连接时间高于 6h。

在理想情况下，Napster 希望它的每个用户都能在下载的同时提供一些文件上传，这样的网络才高效、平衡。然而，在实际的 Napster 网络中，很多用户都是自私的，约 20%～40%的用户几乎从来不提供文件共享而只是下载别人的文件，即

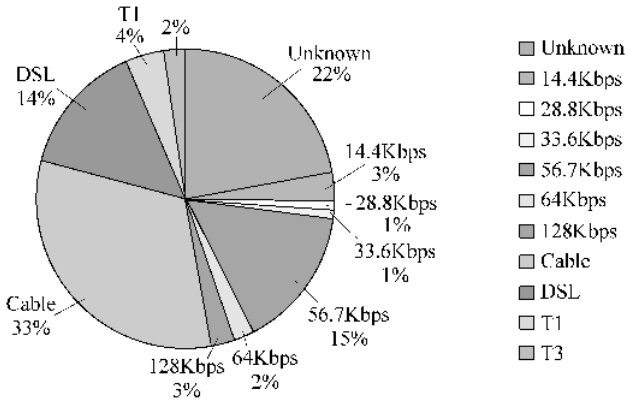

图 2.1.2　Napster 用户所报告的连接带宽和连接设备
（图片来自[Saroiu et al.，2003]）

使对于不自私的结点，绝大部分也只提供少量的文件共享。相反，Napster 中真正提供文件共享的，是网络中少数的约 1%的结点，它们才是 Napster 真正的文件提供者。上述现象在 P2P 领域中也被称为“Free Riding”（搭便车）。

对 Napster 网络性能，我们做如下总结：

（1）Napster 网络中的用户功能上平等，但实际能力差异很大，结点异构性表现在连接带宽、时延、连接时间、共享文件数等多个方面。

（2）许多 Napster 用户不报告或者错误地报告它的连接带宽给 Napster 服务器，这不利于整个网络的高效工作。

（3）Napster 网络中存在很多自私结点，它们只下载文件，几乎从来不上传。自私结点对于整个网络来说几乎是没有贡献只有消耗的。

基于上面的分析，Napster 或者类 Napster 的 P2P 网络在设计、优化时必须考虑这样三个问题：对结点异构性，系统应该有机制针对不同的结点能力采取不同的措施，让它们扮演不同的角色；对用户信息的报告，系统应该有方法鼓励正确的报告或者限制错误报告；对自私结点，系统必须有方法鼓励上传，有限制或禁止自私结点使用网络的权利。

2.1.4　Napster 的陨落和它的现状

Napster 创造了计算机网络领域从未有过的奇迹，然而，这个奇迹并没有持续太久。Napster 被多家唱片商以侵犯版权为由推上被告席而再次成为业界的焦点，最终以 Napster 的陨落作为落幕。下文描述了 Napster 的陨落。之所以要详细描述这一过程，其目的不仅在于 Napster 本身，更重要的是告诉人们 P2P 网络所面临的最大困境：版权问题的法律纠纷。

可能从来没有一个行业像唱片业这样，生存会因为一个小小的软件而受到如此沉重的威胁。1999 年 10 月 31 日，原告之一 BMG 公司和 Napster 达成和解协议，舆论认为这表明唱片业意识到通过数字方式发布音乐将是不可阻挡的潮流，消灭 Napster 也无法阻止其他模仿 Napster 的服务商出现，因此还不如与之合作改变 Napster，将 Napster 的 5000 万用户变成自己的客户，将之变成在线音乐销售的渠道。但随之而来的困境是如何防止那些 Napster 的模仿者继续免费提供歌曲。

1999 年 12 月 7 日，美国唱片业协会(RIAA)代表环宇音乐、索尼音乐、华纳音乐、百代唱片、BMG 等七大唱片公司以违反版权保护法为由把 Napster 公司推向法庭。他们称 Napster 向网民提供 MP3 文件共享软件侵犯了音乐版权，要求法院关闭该公司并判其赔偿损失 1 亿美元。

2000 年 2 月，美国旧金山第九巡回上诉法院的三名法官就音乐网站 Napster 版权纠纷案做出裁决，认为它侵害了各大唱片公司的版权，并且认为 Napster 一直明白它的用户在侵权却对这种侵权行为采取忽视和纵容的态度。但是三名法官并没有应唱片公司的要求，决定立即关闭 Napster 网站，而是把最初的判决送回给低一级的地方法院。法官们说，这项裁决内容过于复杂，需作进一步澄清。错综复杂的法律过程和长达 58 页的判决书反映了现在既缺乏与互联网相关的版权法，也缺乏相应的司法实践。

但是，不管 Napster 案的结果如何，都不能改变 Napster 背后的技术和思想给互联网带来的影响："魔鬼"已经钻出了"魔瓶"，而"魔瓶"也已经被打破了。对于唱片界来说，至少它们销售唱片的方式被彻底改变了。下一个可能是好莱坞的电影工业，因为压缩技术和宽带网络将使得人们能在网络上轻易地传输整部电影，视频和音频就数据量而言已不存在鸿沟。另一方面，网络上已经出现了许多试图推出"合法化的 Napster"模式的公司，譬如由 Napster 的创办人之一创办的 Lightshare.com 和 Flycode.com，利用一个集中的站点提供收费的音乐下载。传统方式下用户每次从网站下载的时候，网站都必须向电信部门交流量费。使用 P2P 则使得这部分费用不再存在了，因此网站获得更多的利润，而用户也可以得到更便宜的音乐。

2000 年 5 月 5 日，美国地区法官 Marilyn Hall Patel 做出判决，依据《美国 2000 年数字版权法》，Napster 的"安全港"资格被取消。

2000 年 6 月，美国唱片业协会(RIAA)和美国音乐出版协会(NMPA)向加利福尼亚州北地区联邦地方法院起诉 Napster 公司，请求法院禁止在社会上流通 Napster 公司的 MP3 文件交换软件"Napster"。

2000 年 7 月 11 日，参议院围绕 Napster 展开的诉讼召开听证会，无果而终。一些议员敦促国会立法，以澄清 Napster 公司是否违反了知识产权法；而支持 Napster 一方的人却认为国会不应该现在就介入双方的争端，以免影响新技术的

发展。7月26日，法官Patel同意美国唱片业协会的要求，做出初步判决，命令Napster立即停止服务。7月28日，美国第九巡回上诉法院暂缓了较低一级法庭的禁令，认为Patel的判决会引发大量问题。

2000年10月31日，Napster宣布同德国的媒体巨人贝塔斯曼集团结成伙伴关系，共同开发基于会员制度的音乐发放系统，通过这种方式，可以保证对艺术家的付款。根据双方的协议，贝塔斯曼同意撤消对Napster的起诉，Napster可使用贝塔斯曼的音乐。

2001年2月12日，美国第九巡回上诉法院做出决定，Napster必须终止其免费互联网服务，并且不得再向乐迷提供共享受版权保护的音乐的服务。

2001年2月20日，Napster提出支付10亿美元的费用给唱片公司以结束诉讼，但是两天后这一提议被唱片公司拒绝。

2001年3月2日，Napster公司宣布，将开始阻止用户下载大约100万首受版权保护的音乐曲目，以遵守法官Patel随时可能下达的一项新的法律禁令。

2001年3月6日，美国地区法官Patel做出判决，责令Napster在5个工作日内删除所有存在争议的歌曲。

2001年6月25日，Napster和英国独立音乐协会、独立音乐合作协会签署了一个世界性的专利使用权转让协定，以给它新的订阅服务提供音乐。

2001年6月27日，电影艺术和科学研究院上诉Napster，控告Napster的在线服务允许用户下载奥斯卡电视广播期间艺员的表演录像。

2001年7月12日，法官Patel命令Napster继续关闭直到它可以证明能够有效地阻止对版权文件的使用。Metallica和Dr. Dre澄清了对Napster的法律争论，结束了双方之间所有的诉讼。

在最兴旺的时候，Napster拥有超过6000万使用网站服务下载歌曲的用户。但是在联邦法院判决文件交换服务违反版权法后，Napster在2001年关闭了服务，最终在2002年6月宣布破产。实际上当贝塔斯曼停止收购Napster时，Napster就已经意识到了这一点，当时的Napster网站上贴出了悲观的图片（见图2.1.3）。

2002年11月，消息传出：美国软件生产商Roxio将以超过500万美元的价格收购目前处于困境的音乐交换网站Napster的剩余资产。在得到特拉华州破产法庭的许可后，这笔交易迅速达成，Roxio以现金外加30万美元Roxio普通股完成交易，获得Napster的专利和品牌标签。但是Roxio不承担Napster的任何债务，Napster的其他硬件资源，诸如服务器、路由器、计算机等仍将是破产企业的一个组成部分，并于12月11日拍卖。对于Napster而言，作为一家软件公司而不是媒体集团，被Roxio收购后，其音乐P2P交换的生涯很难继续下去，实际上真正的Napster的历史已经结束。）

图 2.1.3 陨落的 Napster
（图片来自已关闭的 Napster 网站）

现在，Napster 变成了一个普通的付费音乐下载网站（见图 2.1.4）。虽然名字没有变，网址也没有变（仍然是 http://www.napster.com），但是，这个网站再也不是曾经的 Napster 了，它和 P2P 也再没有任何关系。

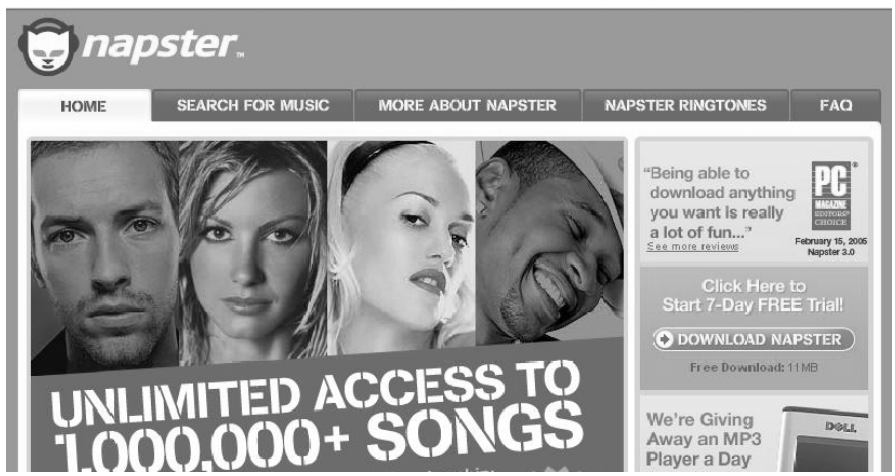

图 2.1.4 2006 年 Napster.com 网站页面

2.1.5 Napster 的缺陷和新的混合式 P2P 网络

1. Napster 的缺陷

（1）C/S 的残余：Napster 网络是 C/S 与 P2P 模式混合的结构，虽然本质的文件交换是 P2P 的，但是文件查询和系统维护都依靠 Napster 服务器，也就不可避免地带来了系统瓶颈、服务器单点失效、可扩展性低等问题。

（2）过于松散的组织管理：Napster 赋予它所有的用户平等的功能，却没有考

虑到它们能力上的差异，同时系统缺乏鼓励用户报告正确的信息和提供文件与别人共享的机制，使得 Napster 不能真正高效地工作。

（3）版权问题：这是 Napster 和整个 P2P 领域面临的最大困境，也是导致 Napster 陨落的直接原因。虽然 Napster 网站服务器里没有任何音乐文件，只是提供文件索引，但是在漫长的法律纠纷之后，Napster 还是被判侵犯版权并从此消失。

2. 新的混合式 P2P 网络

Napster 像一颗流星，它短暂的生命放射出巨大的光芒，虽然陨落却照亮了整个 P2P 领域。在 Napster 之后，新的混合式 P2P 网络快速成长起来，其中最著名的是 BitTorrent，它也是一个以服务器为核心提供共享文件索引的 P2P 网络，所不同的是它提供了文件分片和散列函数映射，同时限定了用户在下载的同时必须上传，而网络和用户信息的更新，尤其是 BT 种子的维护，是依靠服务器中的 Tracker（追踪者）来完成的。下载同一文件的用户围绕 Tracker 形成一个独立的子网。BitTorrent 的服务器基于文件而不是基于整个网络，不同文件的 Tracker 通常在不同的服务器上，从而将服务器分散化了。BitTorrent 是 P2P 在中国最成功的应用，也拥有最为广大的用户群，鉴于此下一节将对 BitTorrent 做深入的讲解。

2.2　BitTorrent——分片优化的新一代混合式 P2P 网络

2.2.1　BitTorrent 的曲折历史

BitTorrent 的第一个可用版本出现在 2002 年 10 月，然而，它的发明人，“BT 之父”Bram Cohen 那时却穷困潦倒。本书对人物的介绍很少，但在这里仍然要不吝篇幅贴出这位值得敬佩的 P2P 技术专家的肖像，见图 2.2.1。幸运的是，BitTorrent 引起了著名免费软件企业家 John Gilmore 的注意，他帮助 Cohen 解决了部分生活费用，使得 BitTorrent 免遭夭折。

图 2.2.1　“BT 之父”Bram Cohen

BitTorrent 真正流行起来是在 2003 年初，它被用来发布一个新版的 Linux，与此同时，还有一些日本卡通的爱好者借助它来共享动画片。Internet2 主干网（又称 Abilene 主干，它连接起了美国 200 多所规模最大的大学，速度比现时的 ADSL 要快 3500 多倍）基础构造的主管 Steven Corbato 说：2003 年 5 月 BitTorrent 的流量开始激增，从 10 月开始 BitTorrent 的流量更是超过了

这个超高速网络总体流量的10%；与之对比，其他文件交换系统的流量没有一个能超过Internet2总体流量的1%。然而，这只是开始。

具有讽刺意味的是，尽管BitTorrent获得了如此的成功，它并没有为Cohen带来一分钱。而在2003年9月，他还在利用一张信用卡的免息期透支来填补另一张信用卡的帐单来过活。

“每个人都有时来运转的时候。”不久，他的事情为Valve软件公司的常务董事Gabe Newell所获悉，尽管Valve正在开发令游戏玩家望眼欲穿的Half-Life 2，但它同时也在建立一个名为Steam的在线分发网络。由于Cohen掌握这个领域的专门技术，所以Valve为他提供了一个职位，Cohen从10月份起搬到西雅图，开始了他的工作。

今天，BitTorrent既指一个混合式P2P网络以及它所对应的协议，同时又是一个支持该协议的应用软件（支持BitTorrent协议的应用软件还有BitComet、BitSpirit、FlashBT等），在BitTorrent.com可以下载到BitTorrent软件不同平台的版本，同时这个网站也是一个非常好的.torrent文件（BT种子）搜索引擎，见图2.2.2。

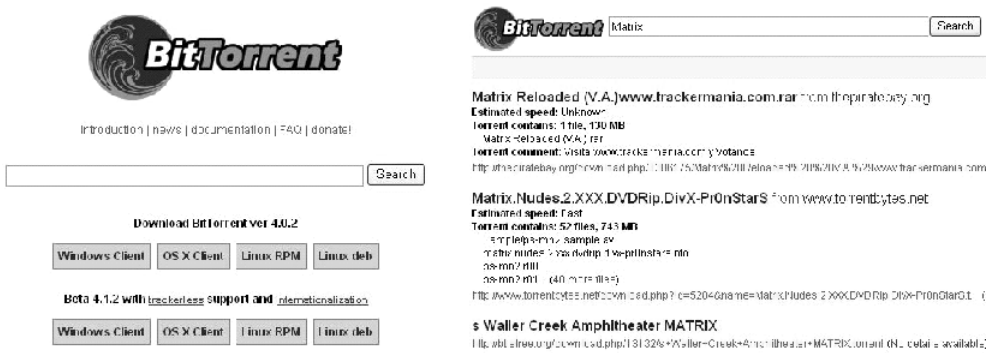

图 2.2.2 www.bittorrent.com及其搜索“Matrix”的结果（2006年12月）

BitTorrent在中国非常流行。在国内所有的P2P应用体系中，BitTorrent占据了大半的市场和用户群。2006年底发布的中国互联网统计报告显示：中国1.11亿网民中有27.8%的人使用过BT软件（超过3000万人），其流行程度由此可见一斑。BitTorrent在中国的风靡一时，实际上创造了P2P领域新的奇迹，它在中国的影响力不亚于P2P的祖先Napster当年在美国带来的巨大冲击。国内常用的BT软件有BitTorrent、BitComet、BitSpirit、FlashBT等，BT网站和搜索引擎已经很多并且一直在扩展中，目前最有影响力的是BT@China联盟（www.btchina.net）、冰鱼BT站（bt.icefish.org）、影视帝国（bt2.cnxp.com）、教育网BT总站（bt.5qzone.net）等。“好123网址之家”网站收录了很多BT网站链接：http://www.hao123.com/bt.htm。

2.2.2　BitTorrent 体系原理

这一小节对 BitTorrent 体系原理的讲述基于 Pouwelse 等人的论文[Pouwelse et al.,04],和他们在 IPTPS'05 上发表的论文[Pouwelse et al.,05]。BitTorrent 网络由 4 个部分组成：BT 网站、.torrent 文件服务器、Tracker、BT 用户,如图 2.2.3 所示(BT 网站未画出)。

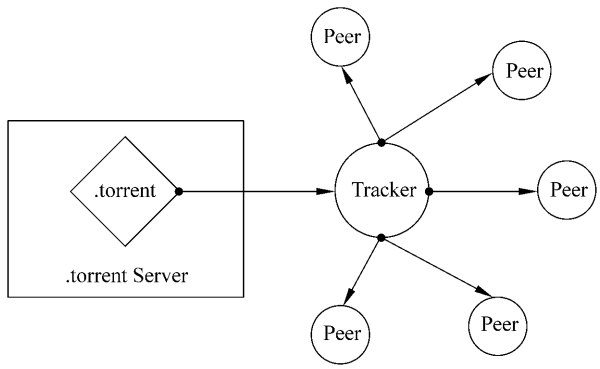

图 2.2.3　BitTorrent 体系原理示意图
(图片来自[Lua et al.,2004])

BT 网站是一系列提供 BT 种子文件(即.torrent 文件)搜索的服务器,从一个 BT 网站可以搜索到一部分种子文件(但这只是因特网 BT 种子文件的一个子集)。由于 BitTorrent 软件本身不提供文件搜索功能,BT 用户要下载文件必须首先到一个 BT 网站进行搜索,从 BT 网站返回的.torrent 文件列表(如图 2.2.4)中选择所要的种子文件。文件列表的含义如下(从左至右)：Added,.torrent 文件上传日期,Name,文件名,Filesize,文件大小；Seeds,种子数(这里"种子"指拥有整个文件的用户)；DLs,在 Tracker 注册的下载用户数(Tracker 跟踪所有下载同一个文件的用户)；Quality,软件运行的操作系统；Submitter,上传.torrent 文件的用户；Info,到相关网页的连接。除了上面的内容,一个.torrent 文件还包含它的 Tracker 的位置,以及所要下载文件的散列值用来验证数据完整性。

Added	Name	Filesize	Seeds	DLs	Quality	Submitter	Info
29-C2	Castlevania (NES) completed in 12:23....	37 Mb	5	5	-	anonymous	link
17-C3	Everquest2.Trailer	205 Mb	0	1	Windows	anonymous	-
07-C3	FFX2:FMVS (StrS FROM DVD)	919 Mb	1	14	PS2	T!dus_Beta	link
06-C2	Partition Magic 8 0 español by Nitro...	50 Mb	4	2	N64	anonymous	-
30-11	Quake 1 beaten in 12 minutes, 23 sec...	166 Mb	8	5	Windows	haze	link
29-C2	Rockman (NES) completed in 21:53 by ...	50 Mb	4	3	-	anonymous	link
15-11	S.T.A.L.K.E.R. Trailer 5 hi-res	47 Mb	0	1	Windows	Ezth Nahua	-

图 2.2.4　.torrent 文件列表
(图片来自[Pouwelse et al.,2004])

.torrent 文件服务器(图 2.2.3 中.torrent Server)保存.torrent 文件，它相当于一个小型的种子数据库，通过 BT 网站来搜索。在 BitTorrent 网络中，一个文件的共享开始于.torrent 文件的产生，通常它由文件的拥有者提交给 BT 网站(这个拥有者是该文件的第一个“种子”)，经过检查确保内容无误后被存放到.torrent 文件服务器中。

Tracker(跟踪者、跟踪服务器)是 BT 网络和用户信息的维护者，其职责是帮助用户相互发现对方，下载同一个文件的所有用户围绕 Tracker 形成一个独立的子网，它对一个文件的下载起着关键性的作用。Tracker 跟踪所有下载某个文件的用户，并实时地将这些用户信息发给其中的每一个，它控制着 BT 网络上某个文件(也可能是多个文件)的下载和用户之间的协同工作。Tracker 和用户之间使用一种很简单的基于 HTTP 的协议进行交互：用户告诉 Tracker 要下载的文件、自己使用的端口以及其他信息，Tracker 告诉用户下载同样文件的其他下载者的联系信息，用户可以利用这些信息相互之间建立连接；此外，一些关于下载和上传速率的信息被发送给 Tracker，Tracker 搜集这些信息用于统计。

在最近版本的 BitTorrent 软件中，即使没有 Tracker 也能定位到拥有文件的用户，这是通过基于 Kademlia 的分布式散列表来完成的(Kademlia 是一种实用、容错的结构化 P2P 网络，将在第 4 章对其进行描述)，在“用户列表”中以“DHT Network”的形式表示出来。但是，目前靠无 Tracker 的 DHT 方式定位文件的速度要远远低于有 Tracker 的服务器方式，因此它只能作为 BitTorrent 的一个辅助定位方法。

每个 BT 用户(图 2.2.3 中 Peer)可以同时下载多个文件，对于每一个文件而言，用户通常按照下述步骤来下载：

(1) BT 用户通过某个 BT 网站搜索文件(如 BT@China 联盟网站)，BT 网站将该搜索请求重定向到它的网站镜像(比如 BT@China 联盟就有很多镜像站点)，检索它所知的所有.torrent 文件服务器，返回给用户关于该文件的.torrent 文件列表。

(2) 用户选择列表中的一项或者多项，被选的每个.torrent 文件会启动一项下载任务。通常，BT 软件会根据.torrent 文件信息连接到该文件所对应的 Tracker，从 Tracker 获得当前也在下载该文件的用户信息(Tracker 一般只返回一定数量的用户而不是全部)。

(3) 用户根据上一步从 Tracker 获得的用户信息与他们建立直接的连接，从他们下载文件的一部分分片，同时也向他们提供上传。

(4) 为了保证 BT 用户的高速下载和整个网络的高效工作，一个用户并不总是和最初建立连接的那些用户互相交换文件，而是每隔一段时间从 Tracker 获得新的用户信息建立新的连接，同时采用“阻塞算法”主动地停止那些对自己来说没

有什么利益的连接、建立对自己更好的连接。

在上述步骤的第 1 步和第 2 步中，出于安全性的考虑，BT 网站可能还会要求用户做一项工作：输入网页中以图形表示的随机字符串，然后才能获得. torrent 文件列表或者连接到文件对应的 Tracker。在第 1 步输入随机字符串的目的，一方面是为了防止通信监听（此时随机字符串相当于文件加密的密码）；另一方面也有效地抵制了 DoS 攻击（denial of service，服务拒绝攻击），如果不要求输入字符串，恶意用户可以无限制地向 BT 网站发出查询请求从而使网站服务器过忙而拒绝提供正常服务。在第 2 步输入随机字符串的目的与第 1 步类似。上述 BT 安全机制虽然很不完善而且留下了很多可以攻击的漏洞，但仍然可以称得上是简单、有效的安全措施。对于混合式 P2P 网络而言，这样的安全机制是合情合理的。

2.2.3　BitTorrent 分片机制

本节所讲的 BitTorrent 分片机制，以及下一小节的 BitTorrent 阻塞算法，都来自 BT 发明人 Bram Cohen 的论文[Cohen. 2003]。

为了跟踪每个 BT 用户都拥有些什么，BT 将文件分割为固定大小的分片（典型的大小是 256KB），每个用户必须通知其他下载者他拥有哪些分片。为了验证数据完整性，对每个分片都通过 SHA-1 散列函数计算出它的散列值保存在. torrent 文件中，用户只有在检查了分片的完整性之后，才会通知其他用户他拥有这个分片。

用户不断地从他能连接到的用户那里下载文件分片，当然，即使是建立了连接的用户，有的也并不包含想要的分片，或者还不允许他去下载（关于不允许其他用户从自己那里下载文件分片的策略，被称为阻塞算法，下一节将讲到）。关于文件分片有更高效、更容错的机制如 Erasure Coding（冗余编码）方法，但是也增加了复杂性；BT 简单地让用户公布它拥有的分片会导致不到 10％的带宽开销，却可以有效、可靠地使用用户的上传能力。

分片流水作业：构架在 TCP 之上的应用层协议，例如 BT，很重要的一点是应该同时发送多个请求，以避免在两个分片发送之间的延迟，因为那样会严重影响传输速率。为了达到这个目的，BT 又进一步将每个分片划分为子分片（典型的大小是 16KB），同时，它一直保持几个请求（通常是 5 个）被流水式地同时发送。流水作业选择同时发送的请求数目的依据，是使大多数连接变得饱和以充分利用带宽。

分片选择：选择一个好的顺序来下载分片对提高性能非常重要。差的分片选择算法不仅低效而且使用户之间相互等待浪费了宝贵的带宽资源。在不同阶段，BT 的分片选择策略有所不同：

（1）严格的优先级（一个分片的下载）

分片选择的第一个策略是：一旦请求了某个分片的子分片，那么该分片剩下的所有子分片优先于其他分片的子分片被请求，这样可以尽可能快地获得一个完整的分片。

（2）最少者优先（文件下载中间阶段/平稳期）

对一个下载者来说，在选择下一个要下载的分片时，通常选择的是他所知用户拥有数最少的那个分片，这就是所谓的“最少者优先”，通俗一点讲其实就是“物以稀为贵”。这种策略确保了每个下载者都拥有与他连接的用户们最希望得到的那些分片，从而一旦有需要，上传就可以开始；同时这也确保了越普通的分片越放在最后下载，从而减少了这样一种可能性：某个用户当前正提供上传，但随后他却没有任何被别人感兴趣的分片了（越稀少的分片通常越让别人感兴趣）。

BT 的分片选择充分考虑了经济学的因素，处处从整个系统的性能出发，参与者越多，系统越优化。如果在某次下载中只有一个种子，而且种子的上传能力比大多数下载者都要差，那么，不同的下载者从种子那里下载不同的分片，系统性能就会变得比较好，因为重复下载相同的分片浪费了种子发出更多信息的机会。“最少者优先”使得下载者只从种子处下载新的分片（也就是整个系统中其他用户都没有的分片），这得益于 BT 网络中下载者能够知道其他用户那里已经有了哪些分片（Tracker 的功劳）。

在某些下载中，原始的种子由于某些原因最终关闭，只好由剩下的这些下载者们来负责上传，这样显然会带来一个风险：某些分片任何一个下载者都没有。“最少者优先”也很好地处理了这个问题：通过尽快地在用户中复制最稀少的分片，减少了由于当前种子停止上传所带来的分片缺失风险。

（3）随机的第一个片断（文件下载最初阶段）

“最少者优先”的一个例外是在下载刚开始的时候，此时下载者没有任何分片可供上传，所以需要尽快获取一个完整的分片。但此时最少的分片通常只有某一个用户拥有，所以，下载它可能比下载多个用户都拥有的那些分片要慢得多。因此，第一个分片是随机选择的，直到第一个分片下载完成，才切换到“最少者优先”策略。

（4）最后阶段模式（文件下载最后阶段）

有时候，可能会从一个速率很慢的用户那里请求一个分片，在下载的中间阶段这没有太大问题，但是在下载的最后阶段它却可能潜在地延迟下载的完成。为了避免这种情况，在最后阶段，下载者向他所连接的所有用户们都发送某分片的子分片请求，一旦某个子分片到了，下载者就会向其他用户发送 Cancel 消息，取消对那些子分片的请求，以避免带宽浪费。实际上，这种方法消耗的带宽并不多，但文件的结束部分可以因此下载得非常快。

2.2.4 BitTorrent阻塞算法

BT并不是由Tracker服务器来集中分配资源，每个用户自己有责任来尽可能地提高自己的下载速率。下载者从他可以连接到的用户处下载文件，并根据对方提供的下载速率给予同等的上传回报（“tit-for-tat”，以牙还牙/针锋相对）。对于合作者，提供上传服务，对于不合作的，就“阻塞”对方。所以说，阻塞是一种临时的拒绝上传策略。虽然上传停止了，但是下载仍然继续，当不再阻塞的时候，也不需要重新建立连接。阻塞算法对提高BT性能是非常必要的。一个好的阻塞算法应该利用所有可用的资源，为所有下载者提供一致、可靠的下载速率，并适当惩罚那些只下载而不上传的自私用户。

（1）阻塞算法的经济学背景——帕累托有效

当一个系统中资源配置已达到这样一种境地：任何重新改变资源配置的方式，都不可能使一部分人在没有其他人受损的情况下受益，这一资源配置的状态称为“帕累托最优”状态，或称为“帕累托有效”（Pareto efficient）。在分布式系统领域，寻求帕累托有效是一种本地优化算法，BT的阻塞算法用一种奉献与回报相对应的方式来试图达到帕累托最优。BT用户对那些向他提供上传的用户给予同样的回报，目的是希望在任何时候都有若干个连接正在进行着双向传输。

（2）BitTorrent的阻塞算法（choking algorithm）

从技术层面上说，BT的每个用户一直对固定数量的其他用户保持疏通（通常是4个，疏通即指不阻塞），所以问题就变成了哪些用户应该保持疏通。这种方法要使得TCP的拥塞控制性能能够可靠地饱和上传容量，尽量让整个系统的上传能力达到最大。

对哪些用户保持疏通应该严格根据当前的下载速率来决定。原来的算法对一段长时间的网络传输进行总计，这种方法效果不好，因为根据资源可用或者不可用，带宽会变化得很快。为了避免因为频繁的阻塞和疏通造成的资源浪费，当前的实现是每隔20秒做一次轮询（10+10）：每隔10秒计算一次哪个用户要被阻塞，然后将阻塞状态保持到下一个10秒。时间间隔取10秒是因为这已经足够TCP来进行调整以使传输率达到最大。

（3）最优疏通（optimistic unchoking）

如果只是简单地为那些向自己提供最高的下载速率的用户们提供上传，那么就没有办法来发现那些空闲的连接是否比当前正使用的连接更好。为了解决这个问题，在任何时候，每个用户都拥有一个称为“最优疏通”的连接，不管它的下载速率怎样。每隔30秒，用户重新计算一次哪个连接应该是“最优疏通”。30秒足以让上传能力达到最大，下载能力也相应达到最大。

（4）反对冷落（anti-snubbing）

某些情况下，一个下载者可能被他连接的所有用户都阻塞了，此时他将会保持较低的下载速率，直到通过“最优疏通”找到更好的用户来连接。为了缓解这个问题，如果一段时间过后，从某个用户那里一个分片也没有得到，那么下载者认为自己被对方“冷落”了，于是不再为对方提供上传（除非对方是“最优疏通”）。“反对冷落”常常会导致多个并发的“最优疏通”，从而使得下载速率恢复得更快。

（5）仅仅上传

一旦某个用户完成了下载，它就不可能再通过下载速率来决定为哪些用户提供上传。目前采用的解决办法是：优先选择那些能从他这里得到更高上传速率的用户，或者选择那些此刻刚好被所有人阻塞的用户，以尽可能地利用上传带宽。

2.2.5 BitTorrent 性能分析

文献[Pouwelse et al.，2004；2005]从 5 个方面分析了 BitTorrent 网络的性能，下面部分总结它们的分析结果。

1. 流行性

由于采用分片优化的多项机制，BT 系统是高效、高可扩展的，只要其核心部分——BT 网站、.torrent 文件服务器以及 Tracker 不发生故障，BT 网络的规模将不断扩大，它也会越来越流行。然而，目前上述三种服务器的故障率都比较高，限制了当前的 BT 规模（如图 2.2.5 所示）。可以明显地看到由于服务器故障导致的几处曲线下降（图片来自[Pouwelse et al.，2004]）

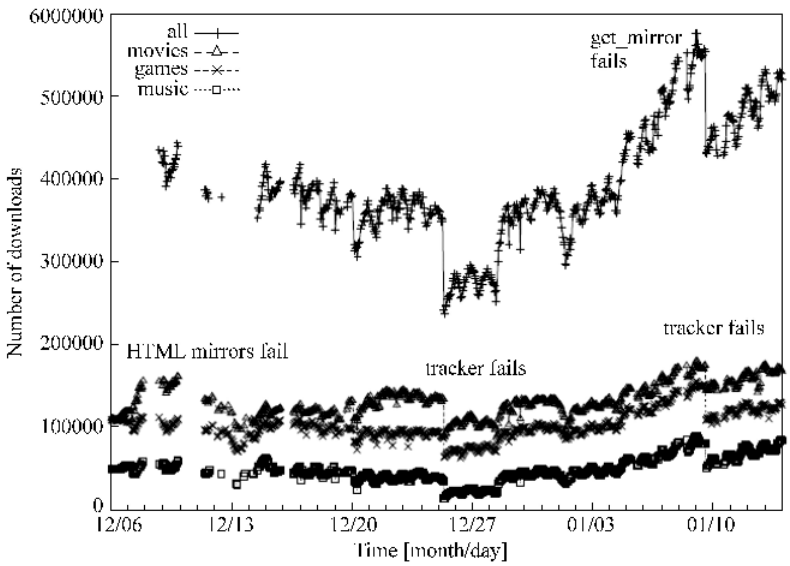

图 2.2.5 一个月的 BT 用户测量（基于 Suprnova 网站）

2. 可用性

BT 的可用性只取决于它的三个核心部分——BT 网站、.torrent 文件服务器以及 Tracker 的可用性，可是目前它们的可用性都不高。总体而言，Tracker 具有较高的可用性，常常可以保证文件被相当一部分用户获得；只有一半的 BT 网站镜像可以正常工作超过 2.1 天；.torrent 文件服务器则比前者更不可靠，如图 2.2.6 所示。

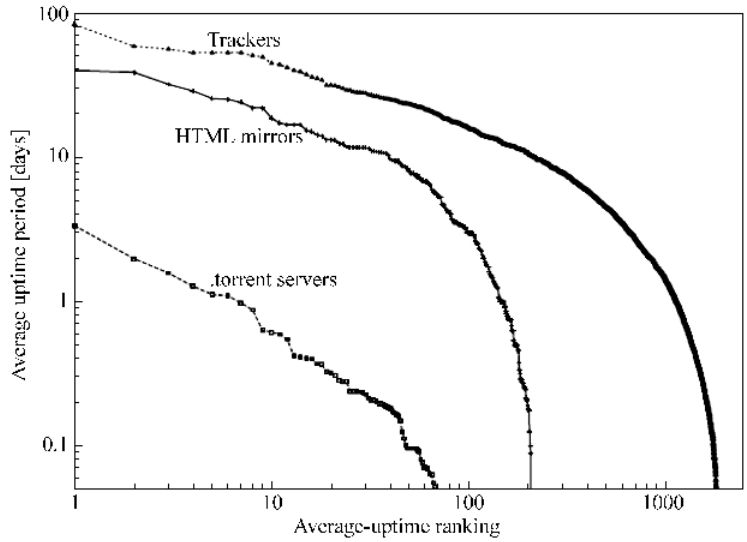

图 2.2.6　BT 网站(HTML mirrors)、.torrent 文件服务器和 Trackers 三者的平均正常工作时间(图片来自[Pouwelse et al.,2004])

3. 下载性能

BT 用户平均的下载速度为 30KB/s，绝大多数低于 65KB/s，这个速度其实还是比较高的，它允许用户可以在一天内下载非常大的文件(比如高质量电影)。从图 2.2.7 可以看出，用户的下载速度和在该速度下的下载数目之间大致符合指数关系。

4. 文件生命期

文件生命期也就是该文件的种子生命期。由于 BT 网络中服务器的故障率较高以及用户(尤其是种子)上传的不确定性，BT 网络的文件生命期很难预测。在一次测量中，发现有 17%的用户在下载完成后做种时间超过 1 小时，仅有 3.1%的用户下载完成后做种时间超过 10 小时。

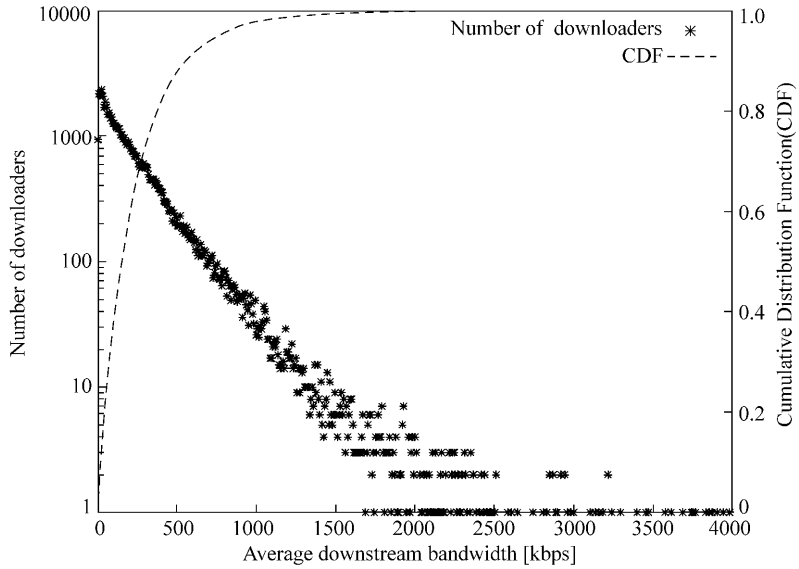

图 2.2.7 BT 用户平均下载速度和在那个速度下的下载文件数目
（图片来自[Pouwelse et al.，2004]）

5. 污染等级

所谓“污染等级”是指 BT 用户插入到 BT 网络中的共享文件的真实性。优越于其他 P2P 应用，BT 有专门的“审查系统”（moderation system）来确保文件内容的真实性以降低污染等级。插入文件的 BT 用户被指派不同的角色：(a)审查者（moderator）：审查别人提交的文件内容，(b)需要审查的提交者（moderated submitter）：普通的不被系统信任的用户，(c)无需审查的提交者（unmoderated submitter）：系统信任的无需审查即可插入文件的用户。如果(b)经常提交真实内容的文件，它可以被升级到(c)，同样如果(c)的表现很好也可以升级到(a)，这非常类似于人类社会的审查制度。鉴于上述等级化的审查系统，BT 的污染等级一般较低，文件真实性是值得信任的。

2.2.6 BitTorrent 体系总结

(1) BitTorrent 是一个混合式结构的 P2P 网络，它以三个部分——BT 网站、.torrent 文件服务器和 Tracker 为核心，控制和帮助 BT 用户共享文件。下载同一文件的用户围绕 Tracker 形成一个独立的子网。

(2) BitTorrent 限定用户在下载文件的同时必须提供上传，这既提高了网络效率，又杜绝了 P2P 网络中的自私结点现象。

(3) BitTorrent 将文件分片，分片又被划分成子分片，子分片流水作业，并且在文件下载的不同阶段有不同的分片选择策略以优化性能。这是 BT 最大的特点，也是它高效的最本质原因。

(4) BitTorrent 基于经济学规律的阻塞算法，优化了网络资源配置，增强了 BT 用户间的协作，它是 BT 网络高效的重要辅助工具。

(5) BitTorrent 对于每个插入的文件有等级化的审查系统，保证了文件的真实性；同时文件、分片都生成对应的散列值供验证，保证了文件的完整性。很多 BT 网站通过要求用户输入随机字符串的方法，有效地防止了通信监听和 DoS 攻击，提供了虽不完善但合情合理的安全机制。

(6) 现有 BT 系统的流行性虽然很高，但是远没有达到其高峰，最主要的原因是 BT 的核心部分——BT 网站、.torrent 文件服务器和 Tracker 故障率高，从而导致低可用性；另一个原因是 BT 中文件的可用性不可预见、不确定，因为这完全取决于“种子”用户的个人喜好，文件无持久性可言。

2.2.7　关于 BT 的一个重要事实澄清：BT 伤硬盘吗

“BT 下载是否伤硬盘?”这是因特网上一个被广泛讨论和关注的问题。因为讨论得多，各种观点鱼龙混杂，反而把一个本来简单的问题说复杂了。BT 下载对硬盘的影响，可以用一句话概括为：不采用磁盘缓存的 BT 下载伤硬盘，采用磁盘缓存的 BT 下载不伤硬盘，而目前的 BT 软件基本都采用磁盘缓存技术，所以通常都不会伤硬盘。

硬盘工作的原理，简单地说就是硬盘磁头读写盘片上的磁道，当磁头读写盘片太过频繁时，盘片可能会由于过热而损坏。最早的 BT 软件没有使用磁盘缓存技术，每下载到一些数据(比如文件的一个分片、一个子分片或者更小的单位)就会把它写入硬盘，当下载量很大时硬盘因为被频繁写入而过热损坏；当上传数据时，硬盘因为被频繁读数据同样会过热损坏。这是“BT 伤硬盘”的原理。

意识到上面的问题后，BT 软件的设计者改变了原有的对磁盘的读写方式，采用了磁盘缓存技术将磁盘读写频率降到非常低，即使 BT 用户高速下载、高速上传也不会伤害硬盘。BT 的“磁盘缓存”技术，对下载而言，每次下载到数据不是直接写入硬盘，而是写入缓存中(缓存通常是内存的一部分)，等到缓存中数据积累到一定容量再整体写入硬盘，因此写硬盘的频率是很低的；对上传而言，需要上传的数据要么可以在缓存中找到、要么可以从硬盘中一次读很多出来，因此读硬盘的频率也是很低的。剩下的问题就是：磁盘缓存应该设多大最合适？通常说来，缓存越大，读写盘次数就越少，也就越保护硬盘。但内存是有限而宝贵的资源，缓存不宜占用太多否则其他软件相应不够用。以 BitComet 软件来说，缓存的默认值在(6～50)MB 之间动态调整，这样的缓存容量已经可以非常有效地保护硬盘了。

2.3 第一代 P2P 网络的特点

2.3.1 拓扑结构

第一代 P2P 网络，无论是 Napster 还是与之类似的 BitTorrent 等，都采用混合式体系结构，即星形拓扑结构，服务器仍然是整个网络的核心。

2.3.2 查询与路由

Napster 的查询机制简单而高效，用户询问服务器，服务器返回所要的文件索引。而 BitTorrent 本身不提供查询机制，BT 网站维护着大量的文件索引（即种子文件），用户到这些网站上去查询文件，本质上这和 Napster 查询差不多。在获得文件索引之后，用户根据文件索引直接建立与文件提供者的连接（在 BitTorrent 中，这之间还有一步与 Tracker 的连接），因此路由跳数是 $O(1)$的，即常数跳。

2.3.3 容错、自适应和匿名性

以服务器为核心的网络，其容错性只在于服务器的故障概率，如果使用多台服务器组成机群，并且提供冗余、替代机制使得一台服务器故障时它的任务可以被其他服务器所分担，那么这样的系统容错性将会非常高。然而，增加、升级服务器通常需要昂贵的支出，一般网站不愿付出此代价，这是 BitTorrent 网络服务器普遍故障率高、可用性低的主要原因。

第一代 P2P 网络的自组织、自适应基本上依靠服务器的监控，用户之间的协作建立在服务器监控的基础之上，因此只要服务器正常工作，网络和结点信息就能得到有效地维护。

可能是出于简单、高效的考虑，目前的混合式 P2P 网络基本上不提供匿名性，但这并不代表混合式 P2P 网络不能提供匿名性。实际上学术界已经有不少论文提出以服务器为核心的匿名方案，而且这些方案也是实际可行的。

2.3.4 增强机制

Napster 是第一代 P2P 网络的代表，但它是朴素的，留下了许多缺陷。在 Napster 的基础上，后起的混合式 P2P 系统都采用了一些增强机制来提高网络的效率，如 BitTorrent 提供文件分片机制，限定用户在下载的同时必须上传以杜绝自私结点的存在，这些都提高了网络工作效率，当然，也增加了网络复杂性。另一方面，在安全性上，BitTorrent 开始逐步采用一些简单、有效的机制防止常见的网络攻击，这相对于 Napster 也是一大进步。